

AdA 223809

INJURY AND COMPENSATION CLAIMS MODULE
MAINTENANCE MANUAL

15-0000000000

D. M. Pearsall

Best Available Copy



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
7100	
A-1	

Medical Decisions Support Department
Naval Health Research Center
San Diego, CA 92138-9174

Report 89-17, supported by the Naval Medical Research and Development Command, Department of the Navy under work unit M0095.005-6304. The views expressed in this article are those of the author(s) and do not reflect the official policy or position of the Department of the Navy, Department of Defense, nor the U.S. Government. Approved for public release, distribution unlimited.

20030210213

AD-A223 809

REPRODUCTION QUALITY NOTICE

This document is the best quality available. The copy furnished to DTIC contained pages that may have the following quality problems:

- **Pages smaller or larger than normal.**
- **Pages with background color or light colored printing.**
- **Pages with small type or poor printing; and or**
- **Pages with continuous tone material or color photographs.**

Due to various output media available these conditions may or may not cause poor legibility in the microfiche or hardcopy output you receive.

☐ **If this block is checked, the copy furnished to DTIC contained pages with color printing, that when reproduced in Black and White, may change detail of the original copy.**

TABLE OF CONTENTS

	Pages
LIST OF FIGURES AND TABLES	iii
SUMMARY	iv
1.0 GENERAL	1
1.1 Purpose of the Maintenance Manual	1
1.2 References	1
1.3 Terms and Abbreviations	2
2.0 SYSTEM DESCRIPTION	4
2.1 System Application	4
2.2 Security/Privacy	4
2.3 Flexibility	4
3.0 SOFTWARE ENVIRONMENT	5
3.1 Programming Language(s) and Conventions	5
3.2 M/FORM and VA File Manager Conventions	5
3.3 M/FORM and Kernel Interface	6
3.4 Security/Privileges	6
4.0 MODULE DOCUMENTATION	7
4.1 Module Design Summary	7
4.1.2 File Overview	7
4.1.3 Naming Conventions	12
4.2 Transaction Documentation (Module Menus)	15
4.2.1 Module Menus	15
4.2.2 M/FORM Documentation	22
4.3 Claims and Compensation Functions	24
4.3.1 Enter/Edit New Claim/Case	24
4.3.2 Enter Claim Forms	48
4.3.3 Continuation of Pay Enter/Edit	53
4.3.4 Compensation Pay Enter/Edit	57
4.3.5 Print Claim Forms	61
4.3.6 Claims Pay-Received Reports	62
4.3.6.1 Pay Information by Employee	63
4.3.6.2 Continuation of Pay	65
4.3.6.3 Compensation Data	67
4.3.7 Tickler File Print/Display	69
4.3.8 Adjudication Status Report	71
4.4 Safety Office Functions	73
4.4.1 Enter/Edit Injury	73
4.4.2 Enter Speedletter	85
4.4.3 Print/Display Safety Data	93
4.4.3.1 Injury Log Report	94
4.4.3.2 Event/Case Summary Report	96
4.4.3.3 OPNAV 5100/10 Log Report	98
4.4.3.4 Personal Case Summary Reports	100
4.4.3.5 Naval Safety Speedletter	103
4.4.3.6 Print Detail of Injuries	107

4.4.3.7	Case Summary for Date/Name Range	109
4.4.3.8	Tickler File Print/Display	111
4.4.3.9	Adjudication Status Report	113
4.5	ICC File Maintenance Functions	115
4.5.1	OWCP Duty Station Agency File - Enter/Edit	115
4.5.2	Reporting Office for OWCP - Enter/Edit	115
4.5.3	Claim Type File - Enter/Edit	116
4.5.4	Nature of Injury File - Enter/Edit	116
4.5.5	Body Part File - Enter/Edit	117
4.5.6	Accident/Injury Type File - Enter/Edit	117
4.5.7	Agent of Accident File - Enter/Edit	118
4.5.8	Cause of Injury File - Enter/Edit	118
4.5.9	Source of Injury File - Enter/Edit	119
4.5.11	OSH Injury Codes File - Enter/Edit	120
4.5.12	Pay Status File - Enter/Edit	121
Appendix A	M/FORM DOCUMENTATION FOR CLAIM FORMS	
Appendix B	FACSIMILE OF CLAIM FORMS	
Appendix C	LIST OF ICC FILE ATTRIBUTES	
Appendix D	ROUTINE DOCUMENTATION	

LIST OF FIGURES AND TABLES

	PAGE
FIGURE 4-1 INJURY COMPENSATION AND CLAIMS MODULE OVERVIEW	8
FIGURE 4-2 INJURY COMPENSATION AND CLAIMS MODULE FILE ORGANIZATION	13
FIGURE 4-3 INJURY COMPENSATION AND CLAIMS MODULE DATA FLOW DIAGRAM	14
FIGURE 4-4 INJURY COMPENSATION AND CLAIMS MODULE CLAIMS FUNCTIONS [T3C INJURY COMP CLAIM MENUS]	18
FIGURE 4-5 INJURY COMPENSATION AND CLAIMS MODULE CLAIMS AND COMPENSATION FUNCTIONS [T3C CLAIMS]	19
FIGURE 4-6 INJURY COMPENSATION AND CLAIMS MODULE SAFETY FUNCTIONS [T3CS SAFETY]	20
FIGURE 4-7 INJURY COMPENSATION AND CLAIMS MODULE MAINTENANCE FUNCTIONS [T2C6 FILE MAINT]	21
FIGURE 4-8 INJURY CLAIMS FORM SELECTION	49
FIGURE 4-9 ILLNESS CLAIMS FORM SELECTION	50
TABLE 4-1 FILE AND GLOBALS USED BY THE INJURY AND COMPENSATION CLAIMS MODULE	9
TABLE 4-2 INJURY AND COMPENSATION CLAIM REFERENCE FILES	10

SUMMARY

The objective of the Naval Occupational Health Information Management System (NOHIMS) is to provide an information system that will aid the Navy in meeting the requirements of the Occupational Safety and Health Act of 1970. In 1983, the Naval Sea Systems Command (NAVSEA) and the Naval Medical Command (NAVMEDCOM) joined to enhance the NOHIMS system. These enhancements are now referred to as the Occupational Safety and Health Record Keeping System (OSHRKS). As a result of these combined efforts, the Injury and Compensation Claims (ICC) module was created.

The ICC module was developed to better manage the Accident reporting system in the Office of Safety and Health (OSH) and to facilitate compensation management in the Personnel Office. The major objectives of this module are as follows:

- Reduce Costs to Navy;
 - Improve Data Consistency/Accuracy;
 - Support OSH/Compensation Reporting;
 - Improve Responsiveness to Inquiries;
 - Support Awards Program.

The major functions of the ICC module are to store data on occupational illnesses and injuries and to generate from these data various statistical reports, including:

- Accident Reporting (Speedletter),
 - Injury and Illness Log,
 - Compensation Reporting using standard forms. (OSHRKS)

The Injury and Compensation Claims Module Maintenance Manual is to be used by the OSHRKS/NOHIMS site managers in conjunction with other Injury and Compensation Claims Module (ICC) technical manuals. The ICC module was created using two packages; VA File Manager and InterSystems' M/FORM. A brief description of each package and the routines used to interface the two packages are discussed in this document. Descriptive flow diagrams, data design tables, and programming code are provided for a better understanding of the ICC Module.

1.0 GENERAL

1.1 Purpose of the Maintenance Manual. This manual is to be used in conjunction with the existing Injury and Compensation Claims (ICC) manuals. This is a technical manual which contains a description of the VA File Manager and InterSystems' M/FORM package interface used to update the ICC module.

This manual describes the software used by the ICC module in the Occupational Safety and Health Record Keeping System (OSHRKS). An overview of the entire module is presented, followed by detailed descriptions of the routines that support each function in the module. This document provides the information needed to maintain the ICC module software. For a management-level view of the ICC module the user is referred to the Injury and Compensation Claims Module Users' Manual. A detailed discussion of how the user interacts with the module is found in the Injury and Compensation Claims Module Operators' Guide.

1.2 References. The following references provide technical and background information about the ICC module:

- o VA File Manager Users' Manual, Version 17, Veterans Administration, March 1986.
- o VA File Manager Programmers' Manual, Version 17, Veterans Administration, March 1986.
- o OSHRKS System-Wide Programmer Maintenance Manual
- o OSHRKS Primer
- o Injury and Compensation Claims Module Maintenance Manual
- o Injury and Compensation Claims Module Users' Manual
- o Injury and Compensation Claims Module Operators' Guide
- o System Manager's Guide
- o System-Wide Programmer Maintenance Manual
- o M/FORM Data Entry Guide, InterSystems
- o M/FORM Programming Guide, InterSystems
- o M/VX System Guide, InterSystems
- o M Programming Guide, InterSystems
- o Injury and Compensation Claims Module Functional Design Specification

- o American National Standard Method of Recording Basic Facts Relating to the Nature and Occurrence of Work Injuries, ANSI Standard Z16.2
- o NAVSEA Instruction 12810.1, Employee Occupational Illness/Injury Compensation Claims; Management and Reporting of
- o Training for Federal Employing Agency Compensation Specialists Resource Book, Federal Employee's Compensation Program U.S. Department of Labor, Employee Standards Administration
- o Navy Occupational Safety and Health (NAVOSH) Program Manual, OPNAV Instruction 5100.23B
- o OPNAV Instruction 5102.19, Mishap Investigating & Reporting

1.3 Terms and Abbreviations. The following terms and abbreviations are used in this manual:

- o **Adjudication:** The process of settling a claim; the final settlement of a claim.
- o **Asterisk:** The notation "*" used to denote a wild card character.
- o **CA Form:** Office of Worker's Compensation Programs (OWCP's) Federal Employee Compensation form.
- o **Code 106:** The Occupational Safety and Health (OSH) Office at the shipyard.
- o **Code 165:** The Employee Relations Office, Compensation Branch at the shipyard.
- o **Controverted:** When an employee's agency believes an employee's compensation claim is invalid, it can indicate disagreement with the claim by controverting it.
- o **COP:** Continuation of employee's regular pay for up to 45 days after traumatic injury.
- o **<CR>:** Notation used to denote a carriage return typically executed by pressing the "ENTER" or "RETURN" key on a terminal device.
- o **DA:** The internal record number variable used by the VA FileMan package.
- o **File Manager:** The data base management package written by the Veterans Administration to interact with the MUMPS programming language. This package, also called FileMan, is used in OSHRKS to handle data manipulation needs.

- o **Global File:** A global or global file is used to retain the data stored in a MUMPS system.
- o **Global Protection:** The ability of the operating system to restrict the access to the global file.
- o **Global Reference:** A subscript which identifies specific location data is stored in a global file.
- o **ICC:** Injury and Compensation Claims
- o **Kernel:** A package of programming utilities written by the Veterans Administration for use with the MUMPS programming language. This package is used in OSHRKS for menu management, task manager, security control, electronic main, and related activities.
- o **MUMPS (Massachusetts General Hospital Utility Multi-Programming System):** A programming language used to develop OSHRKS.
- o **Node:** The information at a specific location in a global file.
- o **Node (Decendent):** An expression used to define the location of information about a subset of a file or record.
- o **Occupation:** A defined set of jobs, each with its own code, that is used to classify the primary function of each employee.
- o **OSI:** Occupational Safety and Health.
- o **OWCP:** Employee Standards Administration, U.S. Department of Labor; organization that adjudicates compensation claims.
- o **Prompt:** A message displayed by the computer in order to elicit a response from the user. The arrow ">" is the standard prompt issued by a MUMPS system.
- o **Stream Format:** A format for data on magnetic tape or VMX files. During input operations this format treats line feeds, form feed, and vertical tabs as string delimiters and ignores carriage returns. During output operations this format packs characters sequentially into the buffer. Instead of splitting strings across block boundaries, this format pads the buffer with null bytes.
- o **System:** The combination of hardware, firmware, and software designed to function as an integrated unit.
- o **Template:** A file containing a description of an input sequence or output report.
- o **Tickler:** A dated "scratch pad" file for noting reminder and/or messages.
- o **VA:** Veterans Administration.

2.0 SYSTEM DESCRIPTION

2.1 System Application. The objective of the Naval Occupational Health Information Management System (NOHIMS) is to provide an information system that will coordinate the components of the Navy's Occupational Health program in order to meet the requirements of the Occupational Safety and Health Act of 1970 and Navy Occupational Health directives. In 1983, the Naval Sea Systems Command (NAVSEA) and the Naval Medical Command (NAVMEDCOM) joined to enhance the NOHIMS system. NOHIMS, whose initial version was developed at the Naval Health Research Center (NHRC), is a composite of two subsystems: an industrial component and medical component. The goal of the enhancement effort was to create a comprehensive occupational health and safety system for Navy industrial facilities, by expanding upon the original NOHIMS functions and adding modules for hazard deficiency abatement, hazardous material control, injury claims and compensation, and safety and health training. These enhancements are referred to as the Occupational Safety and Health Record Keeping System (OSHRKS), were developed using a prototyping approach and the public domain data base management software package, Veterans Administration File Manager. The enhanced NOHIMS contains the following seven modules:

- o Administration
- o Environmental Exposure
- o Hazard Deficiency Abatement
- o Hazardous Materials Control
- o Injury and Compensation Claims
- o Medical Exam Scheduling
- o Safety and Health Training

2.2 Security/Privacy. Security and Privacy Act issues are discussed in the System Manager's Guide and System-Wide Program Maintenance Manual. The Injury and Compensation Claims (ICC) module is not initialized with security but it may be added through the Kernel security functions.

2.3 Flexibility. Since the OSHRKS system was created using the VA File Manager data base utilities, enhancements and updates to the system can be accomplished with minimal programming. The task of enhancing and updating the system has been assigned to the Naval Regional Data Automation Center, (NARDAC), Washington, D.C. Print templates can be created at the individual sites, the exception to this is the ICC module. The ICC module uses the VA File Manager for file design, InterSystems' M/FORM Screen Package for input templates, and the FileMan M/FORM Interface Package. Because custom coding is required to interface these packages, the files and input screens should be modified only by programmers who have received the M/FORM and VA File Manager Interface training.

3.0 SOFTWARE ENVIRONMENT

3.1 Programming Language(s) and Conventions. The ICC module software is written in the MUMPS programming language. MUMPS is a standard language (X11.1-1984) approved by the American National Standards Institute (ANSI), though non-standard dialects exist. Non-standard language features have been avoided as much as possible so that OSHRKS can run in any standard MUMPS environment.

Certain features of OSHRKS, such as error trapping, require the use of implementation-specific language features. When necessary, these features are implemented via M/VX, the InterSystems' MUMPS operating system for the Digital VAX computer. For each option in this manual using non-standard features, an explicit discussion of the feature is provided.

The OSHRKS software uses three MUMPS-based software packages: the VA File Manager data base management system, the VA Kernel system management packages, and the InterSystems' M/FORM screen input package. Knowledge of FileMan is essential to the maintenance programmer. Extensive use is made of FileMan's input, sort, and print templates. Furthermore, ad hoc queries are made through the FileMan Search and Print options. The Kernel package is used in the ICC module to provide security and menu management. M/FORM is used to control inputs into the ICC module. Currently, the forms that have been created using this package have been designed with special coding. Therefore, editing and updating these forms should only be attempted by a MUMPS programmer with knowledge of the M/FORM, FileMan, and Kernel packages. This manual is to be used in conjunction with documentation provided by the Veterans Administration and InterSystems.

3.2 M/FORM and VA File Manager Conventions. The VA FileMan package names variables and routine by beginning them with the letter "D". These "D" variables are not killed by the VA package and should not be killed through any application specific routine that calls a "D" routine. M/FORM uses lowercase letters as its naming convention. Refer to the VA File Manager Programmer Manual and the M/FORM Programming Guide for a list of variables and routines used by both packages.

NOTE: The M/FORM package uses the "NEW" command to clear variables. In doing this, one of the FileMan variables, DA, is cleared, thus causing the user to be logged off of the system. To avoid this error the programmer should edit global node 'ptg(903,ifn,2,n); where ifn = Internal Form Number, 2 = Pre-form code, and n = the line that contains the following code:

```
Old: n filen,rptfile,New,filing,DA s:'$d(nocomp) ...  
New: n filen,rptfile,filing s:'$d(nocomp) ...
```

See the M Programming Guide for an explanation of the "AGED" utility. This utility should be used to change the "Old" code above to the "New" code. After the code has been corrected the form must be recompiled. See the M/FORM Programmer's Guide for an explanation of the 'ptcdo routine.

3.3 M/FORM and Kernel Interface. Transaction processes are invoked through Kernel Menu Options. Most of these option types, are action or print options. The "action" type option executes code to perform a transaction. The code is stored in global reference: `^DIC(19,DA,20)`; where `^DIC(19)` is the Kernel "OPTION" file storage global, DA is the internal number for the menu option, and 20 is the node where the action type options store executable code.

Ex: `^DIC(19,DA,20)='S IFORM=#,CLEAR=1 K DIC(0),WIPE D ^PTRFORM'`

NOTE: Kernel uses the DIM* routines to check embedded code as it is entered. A problem with these routines is that only upper case characters are allowed to be entered. The code shown in the above example would be acceptable to the Kernel, however, not to the M/FORM package. IFORM, CLEAR, WIPE are M/FORM variables that must be entered in lowercase and the ^PTRFORM routine is stored as a lowercase M/FORM routine. Future updates to the Kernel package may make these two packages more compatible, but currently one must enter the action statement as shown above in the example. Then using the ^GED M/VX utility the code should be changed as follows:

EX: `^DIC(19,DA,20)='S iform=#,clear=1 K DIC(0),wipe D ^ptrform'`

Explanation of code:

Set Statement: `iform` = internal form number (# = ifn internal form number) and `clear` set to 1 will erase the screen.

Kill Statement: `DIC(0)` will allow only for M/FORM lookups and `Wipe` will allow the screen to be redrawn.

DO Statements: See M/FORM Programmer's Guide for an explanation of PTRFORM routine.

3.4. Security/Privileges. It is important to note that M/FORM, Kernel, and FileMan have file security. If a file is created in FileMan with a security code assigned, a problem will surface unless the `DUZ(0)` variable is set to the same code before running the M/FORM routine. A similar access problem will occur when the user tries to access the file from an M/FORM input screen. This problem can be corrected by at least three methods:

- o Action Menu options can have a new set command added on each option: `^DIC(19,DA,20)=S iform=#,clear=1,DUZ(0)="0" K DIC(0),wipe d ^ptrform`
- o Security can be deleted from the FileMan Dictionary, `^DIC(FN)`, and added to the M/FORM package
- o Security can be deleted from M/FORM and FileMan packages and added through the Kernel by creating a security key for each module and assigning that key to the individual users.

4.0 MODULE DOCUMENTATION

4.1 Module Design Summary. The ICC module provides information to the Occupational Safety and Health (OSH) Safety Officers and the Compensation Officers at the Naval facilities. These offices use the information generated by this module to help prevent occupational injury and illness, to issue appropriate compensation to Navy civilian employees suffering occupational disability, to minimize the costs associated with occupational injuries and illnesses, to generate descriptive and statistical reports on safety incidents, and to generate compensation claim form facsimilies and statistical reports.

The part of the module illustrated in the left half of Figure 4-1 supports the Safety Office in its investigation and analysis of events and injury reports required by the Navy Safety Center and the Department of Labor (DOL). The Safety Office can record safety incident data using supervisor safety incident investigation reports, employee compensation claim forms, and the shipyard dispensary's daily log. Using interactive computer terminals, the Safety Office staff records injury/illness data in the Injury file using employee information in the Personnel file and a series of module reference files (controlled vocabularies of terms). In addition to supporting the recording of safety incidents, the module also generates safety incident information in several statistical and summary reports.

The part of the module illustrated in the right half of Figure 4-1 supports the Compensation Office in managing and tracking compensation cases, calculating the costs associated with lost worker time, and producing required Department of Defense (DOD), DOL, and OWCP forms and reports. A compensation specialist can record compensation case data using data from an employee's compensation claim form, a physician's medical report, various case update documents, or information gathered from a personal interview with the injured or ill employee. Compensation information associated with the corresponding injury or illness incident data entered by the Safety Office is also stored in the Injury file. The compensation specialist can produce OWCP compensation claim form facsimilies, case summaries, and a number of compensation statistical reports.

4.1.2 File Overview. The ICC module uses several reference files to control vocabularies of terms used in recording data, as well as, files to store individual cases and events. A list of each file is shown in Table 4-1 and a brief description of each is contained in Table 4-2.

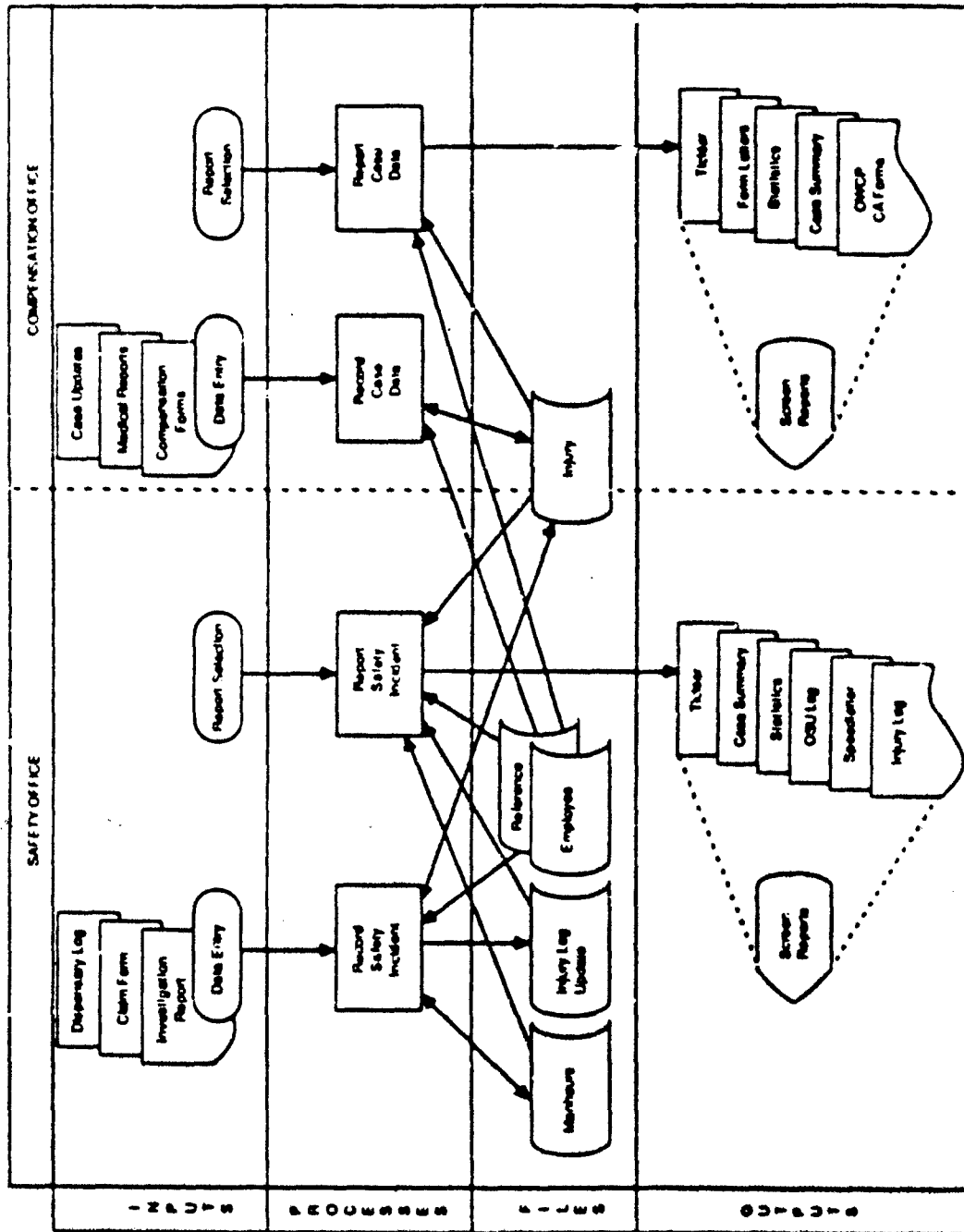


FIGURE 4-1
OSHRKS INJURY AND COMPENSATION CLAIMS MODULE OVERVIEW

TABLE 4-1
FILES AND GLOBALS USED BY
INJURY AND COMPENSATION CLAIMS MODULE

FILE NUMBER	FILE NAME	GLOBAL REFERENCE	MODULE OWNER
1002	Forms	^FORMS(ICC
1005	Injury File	^ICC(ICC
1007	Activities	^INJ(1007,	ICC
1008	COP Data	^INJ(1008,	ICC
1009	Compensation Pay	^INJ(1009,	ICC
1017	Nature of Injury (NOI)	^INJ(1017,	ICC
1018	Body Part	^INJ(1018,	ICC
1025	Accident/Injury Type	^INJ(1025,	ICC
1026	Cause of Injury	^INJ(1026,	ICC
1027	Source of Injury	^INJ(1027,	ICC
1029	Claim Type	^INJ(1029,	ICC
1031	Manhours	^INJ(1031,	ICC
1057	OSHA Injury Codes	^INJ(1057,	ICC
1060	Injury Log Update	^INJ(1060,	ICC
1061	Adjudication Status	^INJ(1061,	ICC
1062	OWCP Agency Codes	^INJ(1062,	ICC
1063	OWCP Reporting Office	^INJ(1063,	ICC
1068	Agent of Accident	^INJ(1068,	ICC
1093	Pay Status	^INJ(1093,	ICC
1004.2	Employee (2000)	^EMPLOY(DA*, "ICC",	ADM
1003	State	^DIZ(1003,	ADM
1074	Agency Unit	^AGENCY(0,	ADM
1001	Occupation	^DIZ(1001,	ADM
1083	Stressor	^STRESS(0,	ADM

*NOTE: DA is the internal storage number of an employee.

TABLE 4-2
INJURY AND COMPENSATION CLAIMS REFERENCE FILES

FILE NUMBER	FILE NAME	DESCRIPTION
1002	Forms	Blank facsimile of each form is stored in this file.
1005	Injury	Contains all information for each individual case or event. The main file for this module.
1007	Activities	A reference file or table listing all activities an injured person is able to perform.
1008	COP Data	Continuation of pay data stored for each individual case.
1009	Compensation Pay	All compensation payments per individual case.
1017	NOI	Categorizes the nature of injury illness suffered by employee.
1018	Body Part	A table of each body part used to control vocabulary.
1025	Accident/Injury Type	A table of accidents and injury types used to control vocabulary.
1026	Cause of Injury	A controlled vocabulary table of injury causes.
1027	Source of Injury	A controlled vocabulary table that is used to categorizes the source of injury/illness.
1029	Claim Type	Includes categories of compensation claims as reported in the Occupational Illness Compensation Claims Report (NAVSEA 12810-1B).
1057	OSHA Injury Codes	Includes injury/illness categories used on the Log of Navy Civilian Occupational Injuries & Illness (OPNAV 5100/10) Report.
1060	Injury Log Update	Used by the Safety Office to report current cases or events.
1061	Adjudication Status	A table that categorizes status conditions in the claims process.
1062	OWCP Agency Codes	Lists codes and textual data describing the federal agencies employing the workforce eligible to file compensation claims to OWCP.

1063	OWCP Reporting Office	Lists compensation offices which process federal compensation claims to OWCP for federal employing agencies.
1068	Agent of Accident	A table that categorizes agent causing accident or injury/illness
1093	Pay Status	A table used to categorize the ways in which employees receive compensation.
1004.2	Employee Claims	The repository of all claim forms filed by the employee and entered by the claims worker.
1003	State	A table of U.S. states used for controlled vocabulary.
1074	Agency Unit	A file created by the Admin. module which lists all organizational units for an agency or facility.
1001	Occupation	The table of occupations defined by the government and used as controlled vocabulary.
1083	Stressor	A table of materials used at the facility that may be a cause of the accident or illness.

The Injury file is the primary repository of information for the ICC module. It contains a separate record of each injury or illness event for each employee who suffers an occupational injury or illness. Additionally, associated with each injury/illness event can be one or more compensation claim forms, which are stored in the Claim subfile in the Employee file. The injury file references the demographic information of a given employee. This information is stored in the Employee file and referenced through a pointer. The identifying field, case serial number, is created when a new case is entered into the Injury file. Each record in the Injury file describes a given injury or illness case for a selected employee. Since an employee can suffer more than one injury or illness over time, each file entry also requires the employee's duty station, assigned shop, and occupation at the time of the event.

Figure 4-2, shows the pointer fields that link the Injury file entries to various controlled vocabulary reference and data files. This diagram shows several Injury file data elements that point to reference files. The Injury file is not referenced by other modules.

The key reference file used by the ICC module is the Employee file. Created and maintained by the Administration module, the Employee file contains demographic and organizational information on employees known to OSHRKS. Every Injury record describing a particular injury/illness event is associated with one and only one entry in the Employee file. Only employees having a record in this file can have an injury record created. A subfile in the Employee file #1004.2 is the repository for all claim forms filed. Once a form has been filed, it can not be edited. Figure 4-3 shows the relationship between the Employee Claims subfile and the ICC module.

4.1.3 Naming Conventions. The ICC module tables are stored in the 'INJ global and case injury/illness data are stored in the 'ICC global. All option names, input templates, sort templates, print templates, and routines begin with T2C or T3C, the number denotes the current versions of this module.

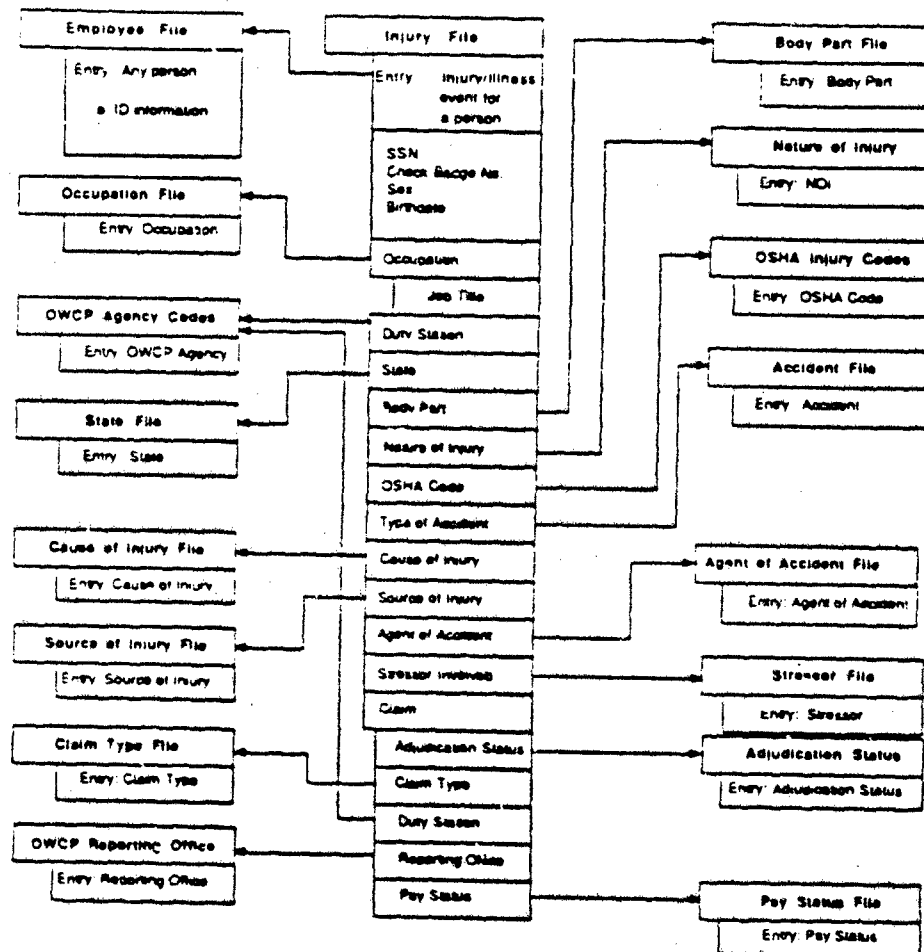


FIGURE 4-2
OSHRKS INJURY AND COMPENSATION CLAIMS MODULE
FILE ORGANIZATION

INJURY COMPENSATION MODULE DATA FLOW

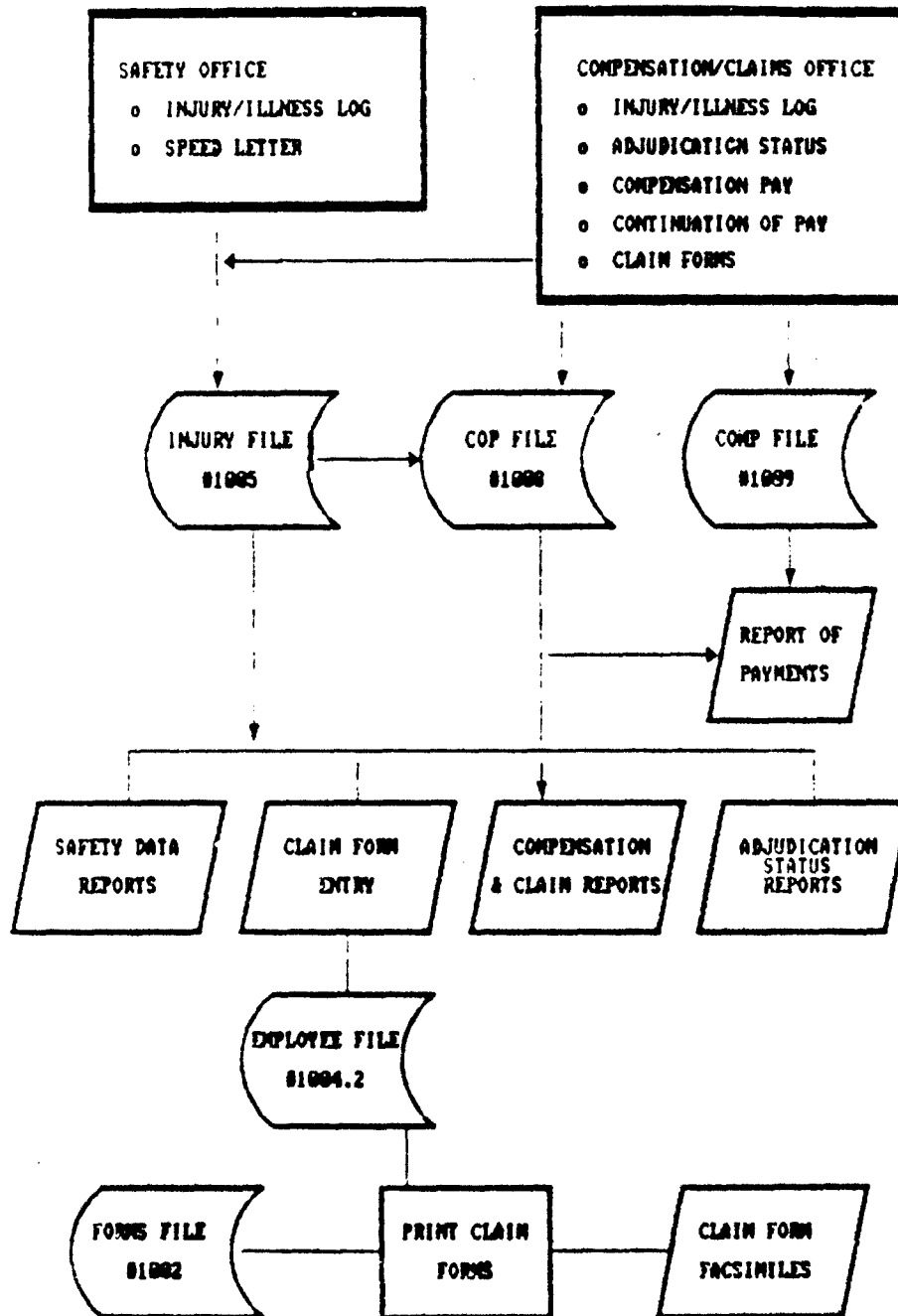


FIGURE 4-3

4.2 Transaction Documentation (Module Menus). The Injury & Compensation Claims Module [T3C INJURY COMP CLAIMS] supports three distinct functional areas; Claims and Compensation, Safety, and Reference File Maintenance. Figures 4-4, 4-5, 4-6, and 4-7 show the transaction flows of this module.

4.2.1 The actual menus for each functional area are created for site specific needs. Reports and options for an individual site will not be discussed in this manual. This section will only discuss the baseline options that are created at the time that the system is installed.

INJURY AND COMPENSATION CLAIMS MODULE MENUS

Injury & Compensation Claims Module (T3C Injury Comp Claims)

- | | |
|--|--|
| 1 Claims and Compensation Functions ——— [T3C CLAIMS] | 1 Enter/Edit New Claim/
Case [T3C NEW CASE] |
| | 2 Enter Claim Forms [T3CLAIM] |
| | 3 Continuation of Pay
Enter/Edit [T3COP] |
| | 4 Compensation Pay Enter/
Edit [T3COMP] |
| | 5 Print Claim Forms
[T3CPFNT] |
| 6 Claims Pay-Recd Reports [T3C ———
CLAIMS PAY DATA] | 1 Pay Information by
Employer
[T3CPAY1 PAY TYPES] |
| | 2 Continuation of Pay
[T3CPAY4] |
| | 3 Compensation Data
[T3CPAY3 COMP DATA] |
| | 7 Tickler File - Print/
Display
[T3CSR8 TICKLER] |
| | 8 Adjudication Status
Report
[T3CSR9 ADJUDICATION
STATUS] |

2 Safety Functions [T3CS SAFETY]-----

3 Select Print/Display Safety Data—
[T3CSR SAFETY REPORTS]

- 1 Enter/Edit Injury
[T3CSLG]
- 2 Enter Speedletter
[T3CSPD]
- 1 Injury Log Report
[T3CSR1 INJURY LOG]
- 2 Event/Case Summary Report
[T3CSR2 CASE SUMMARY]
- 3 OPNAV 5100/10 Log Report
(OSHA 100/F MOD)
[T3CSR3 OPNAV 5100/10
LOG]
- 4 Personal Case Summary
Report
[T3CSR4 PERSONAL CASE
SUMMARY]
- 5 Naval Safety Speedletter
[T3CSR5 SPEEDLETTER]
- 6 Print Detailed All
Injuries Report
[T3CSR6 ALL INJURIES]
- 7 Case Summary for Date/
Name Range [T3CSR7
CASE RANGE SUMMARY]
- 8 Tickler File - Print/
Display [T3CSR8 TICKLER]
- 9 Adjudication Status Report
[T3CSR9 ADJUDICATION REPORT]

3 ICC File Maintenance Functions —————

- 1 OWCP Duty Station Agency
file- Enter/Edit
[T2C6 OWCP AGENCY]
- 2 Reporting Office for
OWCP - Enter/Edit
[T2C6 OWCP REPT OFFICE]
- 3 Claim Type File - Enter/
Edit [T2C6 CLAIM TYPE EDIT]
- 4 Nature of Injury File -
Enter/Edit [T2C6 NOI EDIT]
- 5 Body Part File - Enter/
Edit [T2C6 BODY PART EDIT]
- 6 Accident/Injury Type File-
Enter/Edit [T2C6 ACCIDENT
FILE EDIT]
- 7 Agent of Accident File -
Enter/Edit
[T2C6 AGENT ACCIDENT EDIT]
- 8 Cause of Injury File -
Enter/Edit [T2C6 CAUSE
INJ EDIT]
- 9 Source of Injury File -
Enter/Edit [T2C6 SOURCE
INJURY EDIT]
- 10 Adjudication Status File
Enter/Edit [T2C6 ADJUD
STATUS EDIT]
- 11 OSH Injury Codes File-
Enter/Edit
[T2C6 OSH INJURY CODES
EDIT]
- 12 Pay Status File - Enter/
Edit [T2C6 PAY STATUS
EDIT]

INJURY COMPENSATION CLAIMS MODULE (T3C INJURY COMP CLAIMS)

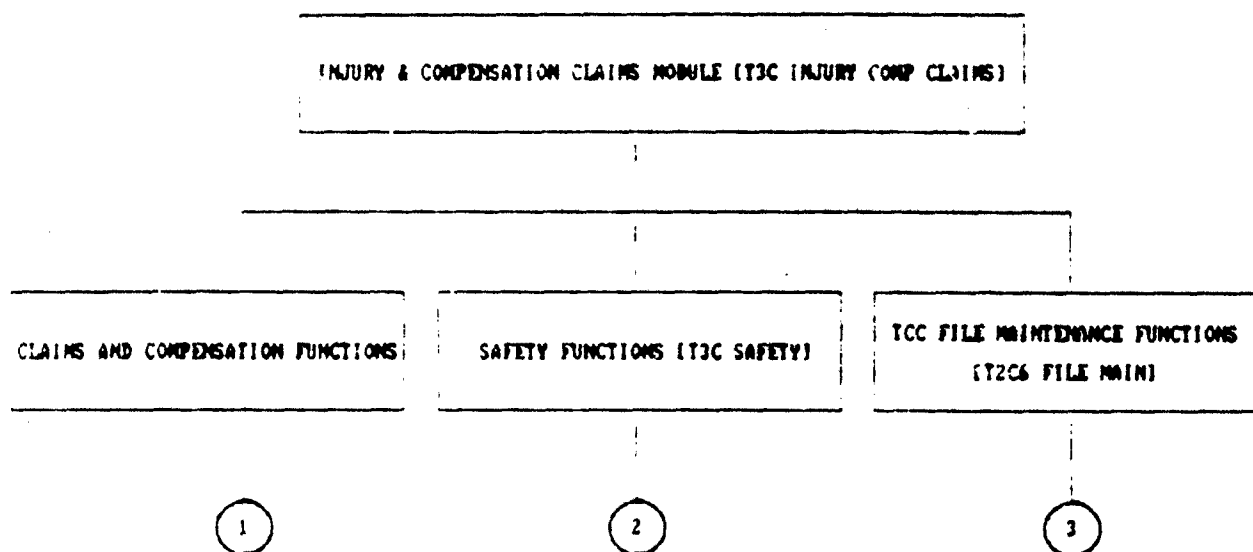


FIGURE 4-4 TOP LEVEL OPTIONS

CLAIMS AND COMPENSATION FUNCTIONS [T3C CLAIMS]

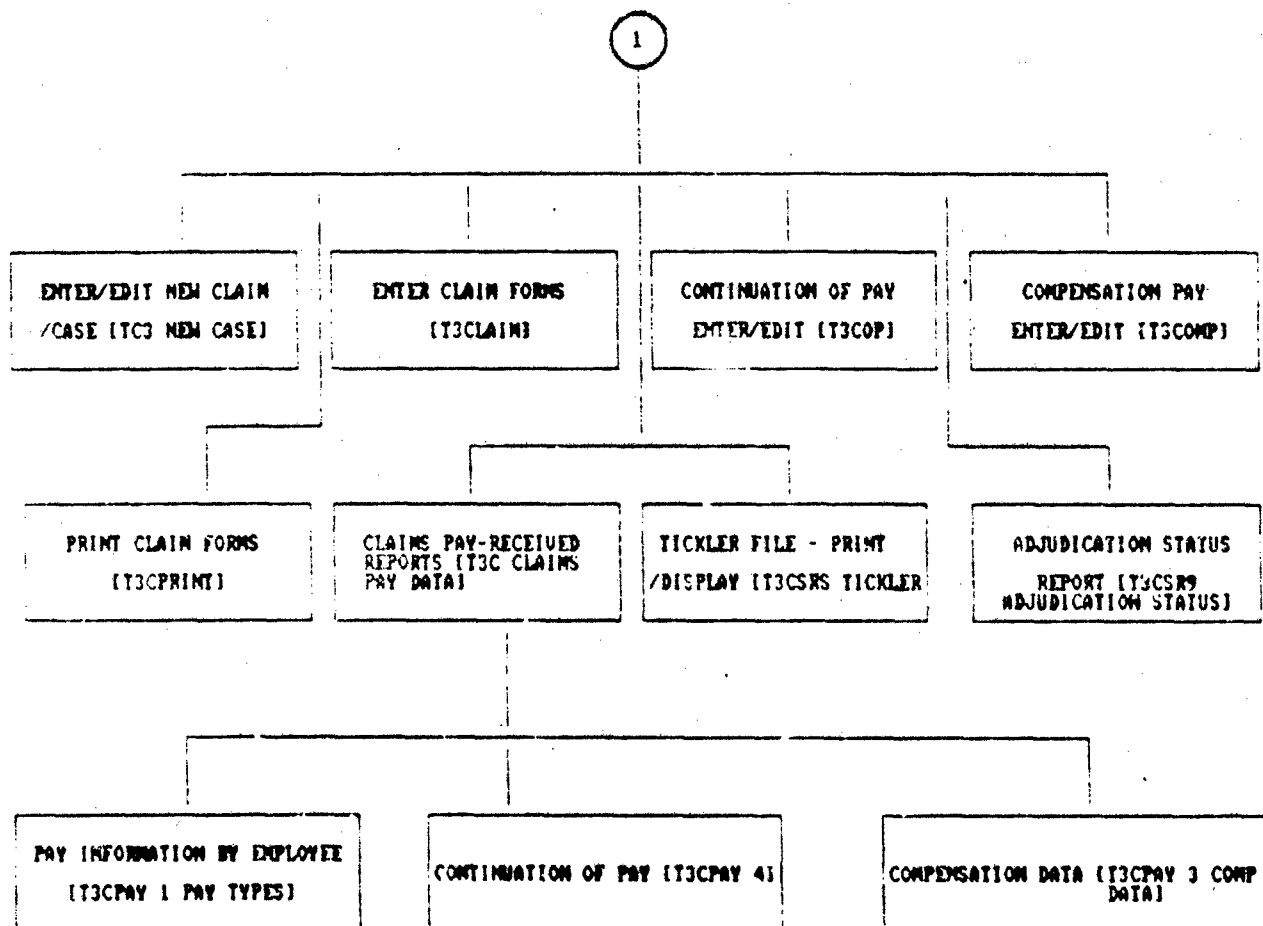


FIGURE 4-5

SAFETY FUNCTIONS [T3CS SAFETY]

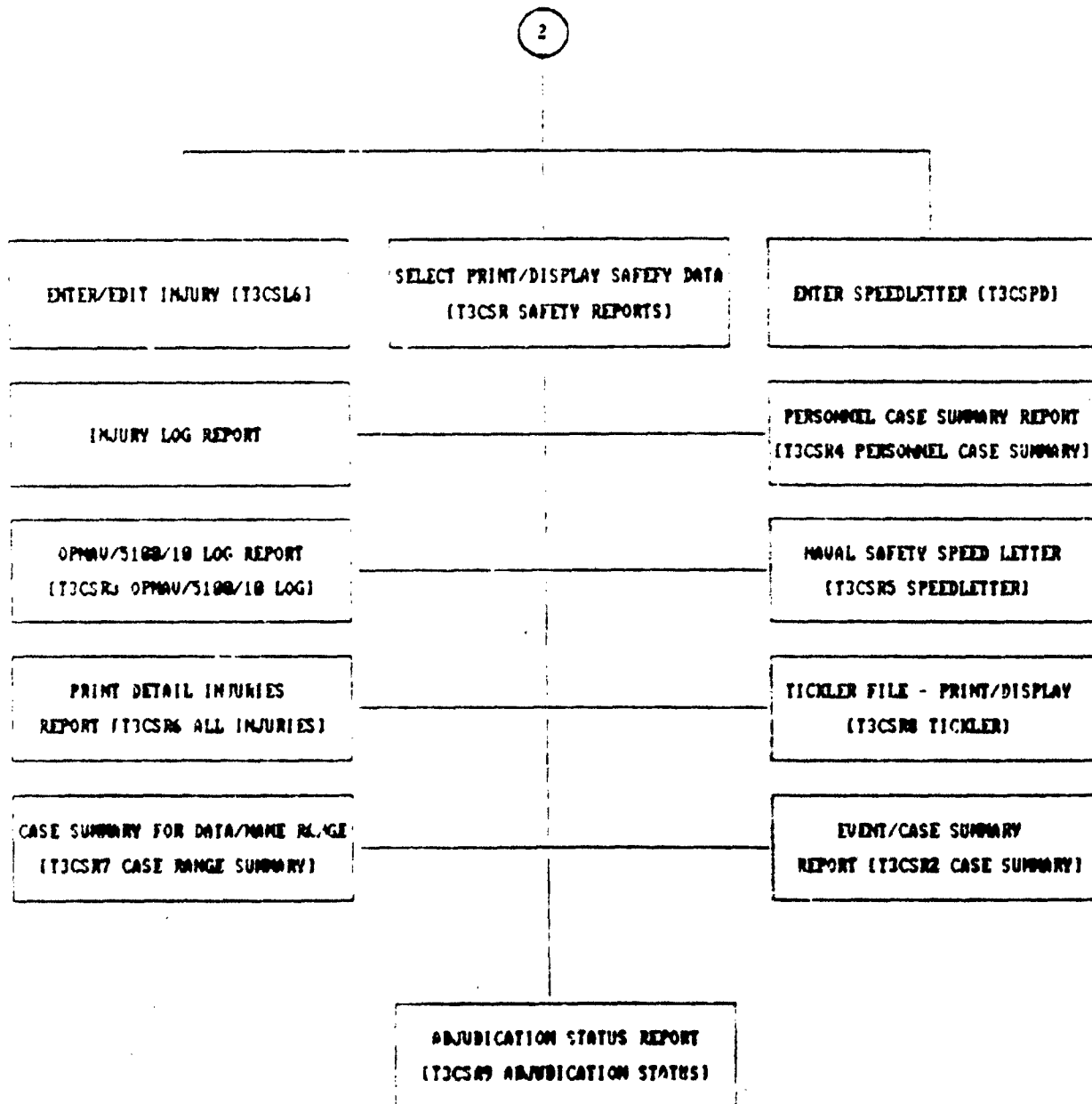


FIGURE 4-6

ICC FILE MAINTENANCE FUNCTIONS [T2C6 FILE MAINT]

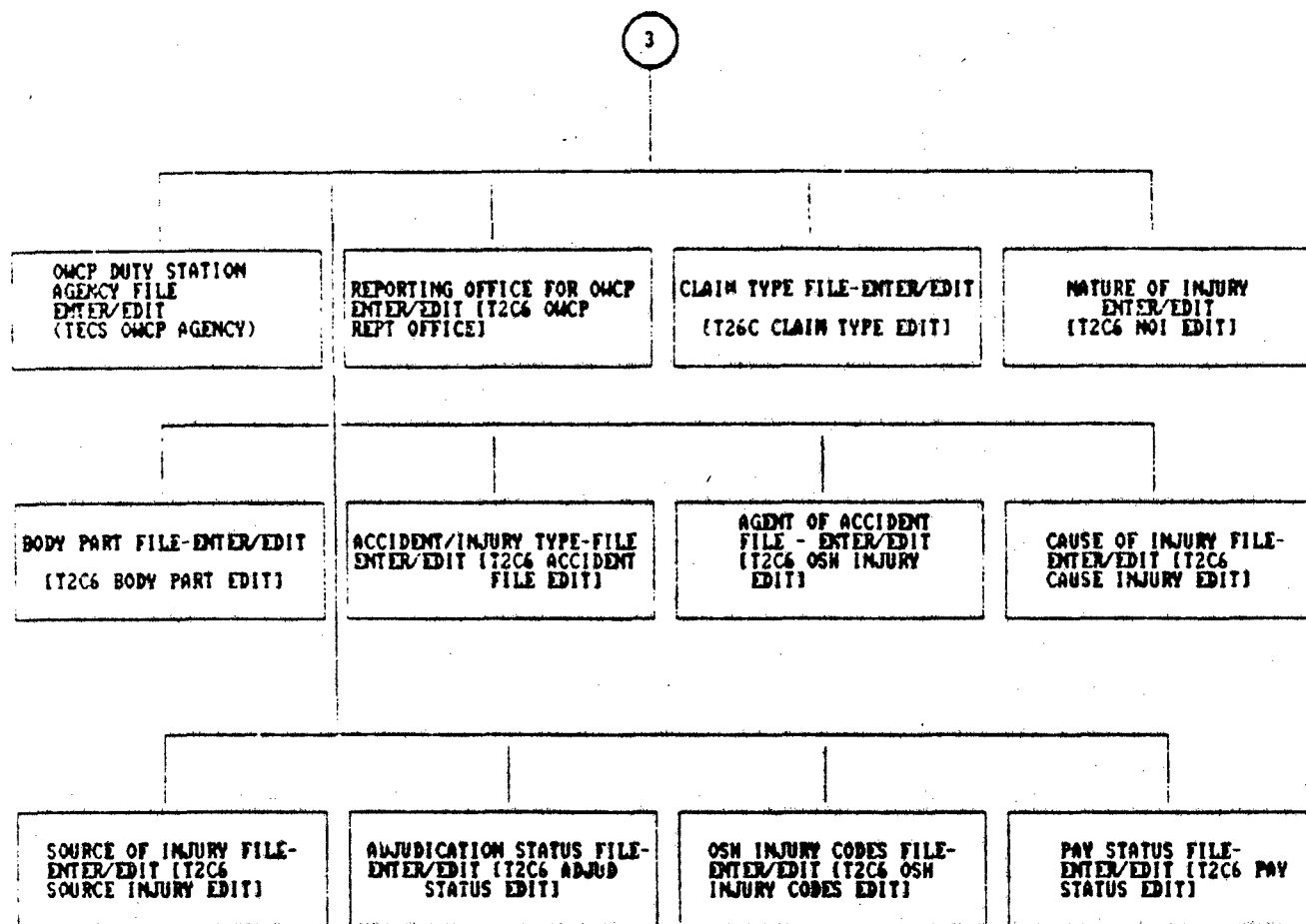


FIGURE 4-7

4.2.2 M/FORM Documentation. The display window is shown for each transaction that calls the ptrform routine. The M/FORM interface documentation (created by running the T3CDOC routine) will follow the windows and describe by the following outline:

FORM NAME AND M/FORM INTERNAL IDENTIFICATION NUMBER

- I. PRE-FORM CODE — Code that is executed prior to displaying the input windows.
- II. WINDOW DOCUMENTATION — For each window shown the window name and internal number will be shown.
 - A. FileMan Data Description of each data element described in the input template.
 1. Internal FileMan data element number (DA) and external FileMan data element name.
 2. Data type as described by FileMan and used by M/FORM.
 3. Pointer Global Reference or Input Verification. If the data element points to another file, the file number will be listed and the global reference will be listed. If the data is not a pointer then the executable checking code will be listed.
 - B. M/FORM Reference Data Description. (Shown under each data FileMan Data Description.)
 1. Internal File Identification number (FID) and data element name.
 2. Internal-External Conversion Code — The M/FORM utility uses "om" and FileMan uses "X" as input variables.
 3. Stored — Actual global where data is stored. The word "nowhere" is used when the field is either computed or that calls another form.
 4. Lookup Node — List of each data element that is displayed in the lookup window.
 5. Index — Cross reference used as a lookup.
 6. Pre-field Code — Code that is executed before the question is asked for a particular element.
 7. Post ID — After the FileMan input verification code is executed the POST-ID code is executed for additional checking.

8. Post-Field Code — Post-Field Code is executed after a question has been answered and inputs are verified to be valid, and prior to filing.
 9. Post-Filing Code — After the response has been filed this code is executed.
- C. Calling Code. For each data element that calls for a form number is shown. The form is outlined in the same manner as described above.
- III. POST-FORM CODE — This code checks to see if the record is filed and then calls the module interface routines (T3C*).

4.3 Claims and Compensation Functions [T3C CLAIMS]. Menus under the Claims and Compensation functions are as follows:

4.3.1 Enter/Edit New Claim/Case [T3C NEW CASE].

Purpose of transaction: This option is used to create new case records, edit existing claim data, update claims and compensation information, and record the current adjudication status.

Transaction Description: This transaction is called from a Menu option [T3C NEW CASE] in the OSHRKS application package.

Type of transaction: Input template # 600 created using M/FORM

Global Dictionary Reference: M/FORM— `ptg(903,600,

Type of Menu Option: Action

Action/Routine Code: s iform=600,clear=1 k wipe,DIC(0) d `ptrform

Output: Stored in Injury File #1005 in `ICC(N ;where N=internal case number.

Retention: Once entered a case may not be deleted without programmer access.

Security/Privacy: Calls `T2USER to verify agency access.

Flexibility: Created using M/FORM-FileMan Interface, should not change form.

Error Recovery: Error trap by Kernel `TER routine

Report Format Requirements: None

Inputs: See attached.

M/FORM #600: INJURY/ILLNESS ENTRY

WINDOW #1 INJURY INFORMATION

INJURY/ILLNESS DATE & TIME _____ CASE NUMBER _____

CLAIM RECORD CREATE DATE _____ INJURY INFORMATION
DATE OF CLAIM _____

CLAIMS CLERK ID _____ CLAIM TYPE _____
WHERE FILED OFFICIAL PERSONNEL FOLDER _____ SENT TO OWCP? _____
DATE CLAIM SENT TO OWCP _____ OWCP FILE NUMBER _____
DATE TRANSFER FEHB TO OWCP _____ FINAL DISPOSITION _____
CLAIM CONTROVERTED _____₁ PRESS <RETURN> FOR CONTROVERSION EXPLANATION

EXTENT OF INJURY _____ LOST TIME ACCIDENT _____
NATURE OF INJURY _____
BODY PART _____
TYPE OF ACCIDENT _____
CAUSE OF INJURY _____
SOURCE OF INJURY _____
INVESTIGATED BY _____ BLDG/SHIP _____
SPECIFIC LOCATION _____
DAYS HOSPITALIZED _____ JOB TRANSFER MED REASONS _____
RETIRED ON DISABILITY _____
THIRD PARTY CASE _____ OUTCOME _____
ADJUDICATION STATUS _____₂ PRESS <RETURN> FOR KEY EVENTS TRACKING SYSTEM

WINDOW #2 MEDICAL INFORMATION

PRIMARY PHYSICIAN _____ TAX ID _____
DOES DOCTOR SPECIALIZE? _____ SPECIALTY _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
PHONE NUMBER _____ DATE SIGNED _____

REFERRAL PHYSICIAN _____ TAX ID _____
DOES REF DOCTOR SPECIALIZE? _____ SPECIALTY _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
PHONE NUMBER _____ DATE SIGNED _____

MEDICAL CARE FACILITY _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
PHONE NUMBER _____

WINDOW #3 EMPLOYEE INFORMATION

INJURY/ILLNESS DATE & TIME _____ CASE NUMBER _____
NAME _____
SSN _____ SEX _____ BADGE NUMBER _____ BIRTHDATE _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
OVERSEAS ADDRESS _____
HOME PHONE NUMBER _____ EMPLOYEE COST CENTER _____
AGENCY _____
MONTHS ON JOB _____ FEGLI CODE _____ FEHB CODE _____
REGULAR WORK SCHEDULE _____ REGULAR WORK HOURS _____
REGULAR NUMBER OF HOURS WORKED PER DAY _____
NUMBER OF PART-TIME HOURS WORKED PER DAY _____
SUPERVISOR'S NAME _____
SUPERVISOR'S TITLE _____

1 WORD PROCESSING WINDOW M/FORM #608: CONTROVERSION EXPLANATION
2 M/FORM #608

_____ KEY EVENTS TRACKING SYSTEM _____
ADJUDICATION STATUS _____
DATE ENTERED _____ DATE REMOVED _____

M/FORM: INJURY/ILLNESS ENTRY (#600)

I. PRE-FORM CODE

```
n filen,New,rptfile,filing s:'$d(New) New=0 s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s DZ(0)="I"
s U- " I '$d(DT) s X=$d(SH),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm
s U-"" i '$d(DT) s X=$d(SH),%DT="" d %DT s DT=Y k X,Y
n filen,rptfile,filing s:'$d(nocomp) filen='ptc(iform,99)
```

II. WINDOWS

WI EMPLOYEE LOOKUP

DATA ELEMENT	DATA TYPE	FOINIER GLOBAL REFERENCE OR INPUT VERIFICATION
1 INJURY/ILLNESS DATE & TI	DATE	S %DT="ET" D %DT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 91

20 CONVERSION CODE:

```
s X=" " s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " $e(cm "000",9,10) ":" $e(cm "000",11,12) s X=X "
s:X="(1700)" X="" s cm=X
```

```
28 STORED: $p("ICC(DA,0),"",2)
32 POST FILING CODE: d setindx`fmfm
54 INDEX: "ICC("D")
55 LOOKUP NODE:
91-23`$p("ICC(Lids,0),"",2)
47-30`$p("ICC(Lids,0),"",3)
97-45`$p("ICC(Lids,100),"",2)
```

M/FORM FID # 117

20 CONVERSION CODE:

```
28 STORED: $p("ICC(DA,0),"",1)
44 POST ID: s:cmcl cm="" s:cm999999999 cm=""
```

2 NAME

FOINIER 1004' EMPLOY(

M/FORM FID # 47

20 CONVERSION CODE:

```
s cm=$s(cm:"":"",('$d("EMPLOY(cm,0))#10):cm,1:$p("EMPLOY(cm,0),"",1))
28 STORED: $p("ICC(DA,0),"",3)
```

30 POSTFIELD CODE:

```

NEW X K P, FLAG, XI, X, AG, CL, CC, %
S U="" I $d(new)&(old'=new) S FLAG="" P(0)=EMPLOY(new,0)
I $D(FLAG) S:$D(EMPLOY(new,1)) P(1)=EMPLOY(new,1)
I $D(FLAG) S:$D(EMPLOY(new,2)) P(2)=EMPLOY(new,2)
I $D(P(0)) S $p(cbi(4),db,1)=SP(P(0),U,2) ;SSN
I $D(P(0)) S $p(cbi(5),db,1)=SP(P(0),U,3) ;SEX
I $D(P(0)) S $p(cbi(6),db,1)=SP(P(0),U,5) ;BIOGE
I $D(P(0)) S $p(P(0),U,6),XI=X X "ID"FUNC",22,1) S cbi(7)=X db XI k X,XI ;B
I $D(P(0)) S $p(cbi(20),db,2)=SP(P(0),U,9) s:cbi(20)'="" CO=SP(cbi(20),db,2) ;
I $D(CC) s:CC'="" SP("IOC(DA,3),U,2)=SP("DIZ(1001,CC,0),U,1),
SP(cbi(20),db,1)=SP("DIZ(1001,CC,0),U,2) ;CC
K AFLAG I $D(P(0)) S AG=SP(P(0),U,17) i AG'="" S:$D(AGENCY(0,AG)) AFLAG=1
I $D(AFLAG) s cbi(49)=SP(AGENCY(0,AG,0),U,1) db AG K AFLAG,AG ;AGENCY
I $D(P(0)) S $p(cbi(50),db,2)=SP(P(0),U,12) AG=SP(cbi(50),db,2) ;PCU
I $D(AG) S:AG'="" SP(cbi(50),db,1)=SP(AGENCY(0,AG,0),U,2) K AG ;AGENCY UNIT
I $D(P(0)) S $p(cbi(22),db,1)=SP(P(0),U,19) ;BAYFLAN
I $D(P(0)) S $p(P(0),U,14) S:X'="" cbi(28)=ptc(600,12,28,15,62,X) db X k X
I $D(P(1)) S $p(cbi(130),1)=SP(P(1),U,2) SP(cbi(9),db,1)=SP(P(1),U,4) ;ADDCI
I $D(P(1)) S $p(P(1),U,5) S:X'="" SP(cbi(10),db,1)=SP("DIZ(1003,X,0),U,1) ;S
I $D(P(1)) S $p(cbi(12),db,1)=SP(P(1),U,6) ;ZIP
I $D(P(2)) S cbi(23)=SP(P(2),U,1) ;BAYGRADE
I $D(P(2)) S $p(P(2),U,5) S:X'="" cbi(21)=ptc(600,12,21,15,62,X) db X k X ;
I $D(P(2)) S $p(cbi(24),db,1)=SP(P(2),U,2) SP(cbi(92),db,1)=SP(P(2),U,4) ;PSR
I $D(P(2)) S $p(P(2),U,3) S:X'="" cbi(93)=ptc(600,12,93,15,62,X) db X k X ;
I $D(DT) S $p(SZD(+SH),DT)="" D "DT S DT=Y
S CL="" S:$D("IOC(DA,200)) CL=SP("IOC(DA,200),U,2) s:$d("IOC(DA,100)) "IOC(DA,100)=""
S:CL="" cbi(125)=SZD(+SH) db DT ;CLAIM RECORD CREME DATE
s P=0 f %:1:1 s P=SO(cbi(P)) q:P="" S:cbi(P)'="" cbi(0,P)="" ;stuff for filing
k FLAG,P,X,XI,% ,CC,AG,CL

```

32 POST FILING CODE: d setindx"frma

54 INDEX: "IOC("C")

55 LOOKUP NODE:

47"30"Sp("IOC(Lids,0),"",3)

91"23"Sp("IOC(Lids,0),"",2)

97"45"Sp("IOC(Lids,100),"",2)

W2 INQUIRY INFORMATION

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
201 CLAIM RECORD CREME DATE	DATE	S DT="E" D "DT S X-Y K:Y<1 X

M/FORM FID # 125

20 CONVERSION CODE:

```

s X="(" s:+Se(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"

```

s:X="(1700)" X="" s cm=X

28 STORED: \$p(ICC(DA,200),"",2)

202 DATE OF CLAIM

DATE

S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 69

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(ICC(DA,200),"",3)

9 PREFIELD CODE: i '\$d(cbi(91)) s skip="D"

143 CLAIMS CLERK ID

FREE TEXT

K:\$L(X)>3!(\$L(X)<1) X

M/FORM FID # 58

28 STORED: \$p(ICC(DA,130),"",12)

44 POST ID: s X=cm K:\$L(X)>3!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

202.5 CLAIM TYPE

FOINDER 1029' INJ(1029,

M/FORM FID # 128

20 CONVERSION CODE: s cm=\$e(cm=""',('\$d("INJ(1029,cm,0))#10):cm,1:\$p("INJ(1029,cm,0),"",2))

28 STORED: \$p(ICC(DA,200),"",12)

200 WHERE FILED

SET OF CODES

OFF:OFFICIAL PERSONNEL FOLDER

OWCP:OFFICE OF WORKER COMPENSA

M/FORM FID # 67

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;" ";"_cm ":";2)," ;",1),cm=\$p(cm,";",1)
COMPENSATION PROGRAM;" ;" cm ":";2)," ;",1),cm=\$p(cm,";",1)

28 STORED: \$p(ICC(DA,200),"",1)

203 SENT TO OWCP

SET OF CODES

Y:YES

N:NO

M/FORM FID # 127

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;" ";"_cm ":";2)," ;",1),cm=\$p(cm,";",1)

28 STORED: \$p(ICC(DA,200),"",4)

204 DATE CLAIM SENT TO OWCP

DATE

S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 71

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",

+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) " " Se(cm "000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,200),""",5)

29 PREFIELD CODE: s:'\$d(cbi(127)) skip="D" i '\$d(skip) s:cbi(127)'["Y" skip="D"]

137 OACP FILE NUMBER FREE TEXT K:SL(X)>L3!(SL(X)<1) X

M/FORM FID # 62

28 STORED: Sp(^ICC(DA,130),""",6)

44 POST ID: s X=cm K:SL(X)>L3!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

142 DATE TRANSFER FEB8 TO CW DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 66

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) " " Se(cm "000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,130),""",11)

207 FINAL DISPOSITION

SET OF CODES

A:ACTIVE

I:INACTIVE

M/FORM FID # 75

20 CONVERSION CODE:

s cm=Sp(Sp("A:ACTIVE;I:INACTIVE;",";"_cm_":",2),";",1),cm=Sp(cm,";",1)

28 STORED: Sp(^ICC(DA,200),""",8)

205 CLAIM CONTROVERSED

SET OF CODES

Y:YES

N:NO

M/FORM FID # 90

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO;",";"_cm_":",2),";",1),cm=Sp(cm,";",1)

28 STORED: Sp(^ICC(DA,200),""",6)

205.1 CONVERSION EXPLANATION

M/FORM FID # 73

28 STORED: nowhere

29 PREFIELD CODE:

I cbi(90)[^n](cbi(90)[^N]) s skip="D"
s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d filecm=ptrfilel k shhfile
i \$d(file),file,New=0 s filing=Se('\$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr^fmcm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=CM,DO=CM f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

16 CALLS M/FORM: CONTOVERSIC EXPLANATION (#608)

M/FORM: CONTOVERSION EXPLANATION (#608)

I. PRE-FORM CODE

n file,rptfile,New s:\$d(nocomp) file=ptc(ifom,99)
+s:\$d(nocomp) file=ptg(903,ifom,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$H),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm

II. WINDOWS

CONTOVERSION EXPLANATION #1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 CONTOVERSION EXPLANATIO	WORD PROCESSING	
-----------------------------	-----------------	--

M/FORM FID # 1

28 STORED: Sp('IOC(DA(1),305,DA(0),",",1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

116 EXTENT OF INJURY

SET OF CODES 1:FATALITY
2:PERM TOTAL DISABILITY
3:PERM PARTIAL DISABILITY
4:NO DISABILITY LIKELY
5:MISSING

M/FORM FID # 126

20 CONVERSION CODE: s

cm=Sp(\$p(",1:FATALITY;2:PERM TOTAL DISABILITY;3:PERM PARTIAL DISABILITY;
4:NO DISABILITY LIKELY;5:MISSING;",",",1),cm=\$p(cm,"",1)

28 STORED: Sp('IOC(DA,110),",",1)

104 LOST TIME ACCIDENT

SET OF CODES Y:YES
N:NO

M/FORM FID # 131

20 CONVERSION CODE: s cm=\$p(\$p("Y: YES;N: NO;",",", "_cm_":",2),",",1),cm=\$p(cm,"",1)
 28 STORED: \$p("ICC(DA,100)", "",5)

101 NATURE OF INJURY CODE POINTER 1017' INJ(1017,

M/FORM FID # 97

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d("INJ(1017,cm,0))#10):cm,1:\$p("INJ(1017,cm,0)", "",2))
 28 STORED: \$p("ICC(DA,100)", "",2)
 32 POST FILING CODE: d setirck`fmfm

100 BODY PART CODE POINTER 1018' INJ(1018,

M/FORM FID # 53

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d("INJ(1018,cm,0))#10):cm,1:\$p("INJ(1018,cm,0)", "",2))
 28 STORED: \$p("ICC(DA,100)", "",1)

112 TYPE OF ACCIDENT POINTER 1025' INJ(1025,

M/FORM FID # 54

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d("INJ(1025,cm,0))#10):cm,1:\$p("INJ(1025,cm,0)", "",2))
 28 STORED: \$p("ICC(DA,100)", "",13)
 32 POST FILING CODE: d setirck`fmfm

120 CAUSE OF INJURY POINTER 1026' INJ(1026,

M/FORM FID # 55

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d("INJ(1026,cm,0))#10):cm,1:\$p("INJ(1026,cm,0)", "",2))
 28 STORED: \$p("ICC(DA,110)", "",5)

126 SOURCE OF INJURY POINTER 1027' INJ(1027,

M/FORM FID # 94

20 CONVERSION CODE: s
 cm=\$s(cm="": "", '(\$d("INJ(1027,cm,0))#10):cm,1:\$p("INJ(1027,cm,0)", "",2))
 28 STORED: \$p("ICC(DA,120)", "",3)
 32 POST FILING CODE: d setirck`fmfm

144 INVESTIGATED BY FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 61

28 STORED: \$p("ICC(DA,130)", "",13)
 44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

113 BLDG/SHIP FREE TEXT X:SL(X)>15!(SL(X)<1) X

M/FORM FID # 56

28 STORED: \$p("ICC(DA,100)", "",14)
 44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

114 SPECIFIC LOCATION FREE TEXT K:SL(X)>40!(SL(X)<1) X

M/FORM FID # 57

28 STORED: \$p(ICC(DA,100),"",15)

44 POST ID: s X=cm K:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

119 DAYS HOSPITALIZED NUMERIC FORMAT K:+X'-X!(X>999):(X<0)!(X>.E1".IN.N) X

M/FORM FID # 124

28 STORED: \$p(ICC(DA,110),"",4)

138 JOB TRANSFER MED REASONS SET OF CODES Y:YES
N:NO

M/FORM FID # 63

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_":",2),"",1),cm=\$p(cm,"",1)

28 STORED: \$p(ICC(DA,130),"",7)

139 RETIRED ON DISABILITY SET OF CODES Y:YES
N:NO

M/FORM FID # 64

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_":",2),"",1),cm=\$p(cm,"",1)

28 STORED: \$p(ICC(DA,130),"",8)

140 THIRD PARTY CASE SET OF CODES Y:YES
N:NO

M/FORM FID # 65

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_":",2),"",1),cm=\$p(cm,"",1)

28 STORED: \$p(ICC(DA,130),"",9)

141 CASE OUTCOME FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 78

28 STORED: \$p(ICC(DA,130),"",10)

29 PREFIELD CODE: S:\$p(cbl(65),db,1)("n"!(\$p(cbl(65),db,1)("N") skip="D"

44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

206 ADDITION SIGNUS PRINTER I

M/FORM FID # 129

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbl(0))>9 s shhfile=l d fileone ptrfilel k shhfile

```

i $d(file),file,New=0 s filing=$s('Sd(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

```

W3 MEDICAL INFORMATION

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

400 ATTENDING PHYSICIAN	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
-------------------------	-----------	------------------------

M/FORM FID # 132

28 STORED: Sp('ICC(DA,400),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""] cm=""

400.8 TAX ID	FREE TEXT	K:SL(X)>10!(SL(X)<1) X
--------------	-----------	------------------------

M/FORM FID # 139

28 STORED: Sp('ICC(DA,400),"",9)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""] cm=""

400.6 DOES DOCTOR SPECIALIZE	SET OF CODES	Y:YES N:NO
------------------------------	--------------	---------------

M/FORM FID # 137

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO",";","_cm_":",2);",1),cm=Sp(cm,"",1)

28 STORED: Sp('ICC(DA,400),"",7)

400.7 SPECIALITY	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
------------------	-----------	------------------------

M/FORM FID # 138

28 STORED: Sp('ICC(DA,400),"",8)

29 PREFIELD CODE: S:'Sd(cbi(137)) skip="D" i 'Sd(skip) s:cbi(137)'["Y" skip="D"]

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""] cm=""

400.1 DOCS ADDRESS	FREE TEXT	K:SL(X)>54!(SL(X)<1) X
--------------------	-----------	------------------------

M/FORM FID # 135

28 STORED: Sp('ICC(DA,400),"",2)

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""] cm=""

400.2 DOCS CITY	FREE TEXT	K:SL(X)>20!(SL(X)<1) X
-----------------	-----------	------------------------

M/FORM FID # 134

28 STORED: Sp("ICC(DA,400),"",3)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

400.3 DOCS STATE

FOURNIER 1003' DIZ(1003,

M/FORM FID # 135

20 CONVERSION CODE:
scm=\$s(cm=""": ""', '(\$d("DIZ(1003,cm,0))#10):cm,1:Sp("DIZ(1003,cm,0),"",1))
28 STORED: Sp("ICC(DA,400),"",4)

400.4 DOCS ZIPCODE

FREE TEXT

K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 136

28 STORED: Sp("ICC(DA,400),"",5)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm(""" cm=""

400.5 DOCS PHONE NUMBER

FREE TEXT

K:SL(X)>14!(SL(X)<7) X

M/FORM FID # 156

28 STORED: Sp("ICC(DA,400),"",6)
44 POST ID: s X=cm K:SL(X)>14!(SL(X)<7) X s:'\$d(X) cm="" s:cm(""" cm=""

400.9 DATE SIGNED

DATE

S NOT="E" D "NOT S X=Y K:Y<1 X

M/FORM FID # 140

20 CONVERSION CODE:
s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) _:" Se(cm "C70",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,400),"",10)

401 REFERRAL PHYSICIAN

FREE TEXT

K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 141

28 STORED: Sp("ICC(DA,401),"",1)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

401.8 RP TAX ID

FREE TEXT

K:SL(X)>10!(SL(X)<1) X

M/FORM FID # 149

28 STORED: Sp("ICC(DA,401),"",9)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

401.6 DOES HEP DOCTOR SPECIALI

SET OF CODES

Y:YES
N:NO

M/FORM FID # 147

20 CLVERSION CODE: s cm=Sp(Sp("Y: YES;N:NO;","_cm_":",2",";"),1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,401),"",7)

401.7 RP SPECIALTY FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 148

28 STORED: Sp("ICC(DA,401),"",8)
29 PREFIELD CODE: S:'SD(cbi(147)) skip="D" i 'Sd(skip) s:cbi(147)'["Y" skip="D"
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

401.1 RP ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 142

28 STORED: Sp("ICC(DA,401),"",2)
44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

401.2 RP CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 143

28 STORED: Sp("ICC(DA,401),"",3)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

401.3 RP STATE POINTIER 1003' DIZ(1003,

M/FORM FID # 144

20 CONVERSION CODE: s
cm=Sp(cm,"",1)('Sd("DIZ(1003,cm,0))#10):cm,1:Sp("DIZ(1003,cm,0),"",1))
28 STORED: Sp("ICC(DA,401),"",4)

401.4 RP ZIPCODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 145

28 STORED: Sp("ICC(DA,401),"",5)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'Sd(X) cm="" s:cm["" cm=""

401.5 RP PHONE NUMBER FREE TEXT K:SL(X)>14!(SL(X)<7) X

M/FORM FID # 146

28 STORED: Sp("ICC(DA,401),"",6)
44 POST ID: s X=cm K:SL(X)>14!(SL(X)<7) X s:'Sd(X) cm="" s:cm["" cm=""

401.9 RP DATE SIGNED DATE S WDT="E" D WDT 3 X=Y K:Y<1 X

M/FORM FID # 150

20 CONVERSION CODE:

S X="(" s:+\$(cm,4,5)
X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$(cm,4,5)) " " s:+\$(cm,6,7) X=X \$(cm,6,7) ", " s X=X (1700+\$(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$(cm "000",9,10) " " \$(cm "000",11,12) s X=X " "
s:X="(1700)" X=" s cm=X

28 STORED: \$p("ICC(DA,401),""",10)

1604 MEDICAL CARE FACILITY FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 151

28 STORED: \$p("ICC(DA,1601),""",1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

1604.1MED CARE STREET ADDRESS FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 152

28 STORED: \$p("ICC(DA,1601),""",2)

44 POST ID: s X=cm K:\$L(X)>54!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

1604.2MED CARE CITY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 153

28 STORED: \$p("ICC(DA,1601),""",3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

1604.3MED CARE STATE POINTER 1003' DIZ(1003,

M/FORM FID # 154

20 CONVERSION CODE: s

cm=\$(cm:"""",',(\$d("DIZ(1003,cm,0))#10):cm,1:\$p("DIZ(1003,cm,0),""",1))

28 STORED: \$p("ICC(DA,1601),""",4)

1604.4MED CARE ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 155

28 STORED: \$p("ICC(DA,1601),""",5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:\$d(X) cm="" s:cm[""" cm=""

WM EMPLOYEE DEMOGRAPHICS

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
3 SSN	FREE TEXT	K:\$L(X)>9!(\$L(X)<9)!'(X?9N) X

M/FORM FID # 4

28 STORED: \$p(^ICC(DA,0),"",4)

44 POST ID: s X=cm K:SL(X)>9!(\$L(X)<9)!'(X79N) X s:'\$d(X) cm="" s:cm["" cm=""

4 SEX

SET OF CODES F:FEMALE
M:MALE
U:UNKNOWN

M/FORM FID # 5

20 CONVERSION CODE: s

cm=\$p(\$p("F:FEMALE;M:MALE;U:UNKNOWN;"",":",_cm":",2),":",1),cm=\$p(cm,"",1)

28 STORED: \$p(^ICC(DA,0),"",5)

5 BADGE NUMBER

FREE TEXT

K:SL(X)>11!(\$L(X)<3) X

M/FORM FID # 6

28 STORED: \$p(^ICC(DA,0),"",6)

44 POST ID: s X=cm K:SL(X)>11!(\$L(X)<3) X s:'\$d(X) cm="" s:cm["" cm=""

6 BIRTHDATE

DATE

S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 7

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5)

X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",":",

+Se(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))

s:\$p(cm,"",2)'="" X=X " at " \$e(cm "000",9,10) " " \$e(cm "000",11,12) s X=X "

s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,0),"",7)

7 STREET ADDRESS

FREE TEXT

K:SL(X)>64!(\$L(X)<1) X

M/FORM FID # 130

28 STORED: \$p(^ICC(DA,1),"",1)

44 POST ID: s X=cm K:SL(X)>64!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

8 CITY

FREE TEXT

K:SL(X)>23!(\$L(X)<1)!'(X7AP) X

M/FORM FID # 9

28 STORED: \$p(^ICC(DA,1),"",2)

44 POST ID: s X=cm K:SL(X)>23!(\$L(X)<1)!'(X7AP) X s:'\$d(X) cm="" s:cm["" cm=""

9 STATE

FOUNDER 1003' DIZ(1003,

M/FORM FID # 10

20 CONVERSION CODE: s

cm=\$s(cm:""',"(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,1),"",3)

10.5 ZIP CODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 12

28 STORED: \$p(^ICC(DA,1),"",6)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

9.5 OVERSEAS ADDRESS

FREE TEXT

K:\$L(X)>64!(\$L(X)<1) X

M/FORM FID # 11

28 STORED: \$p(^ICC(DA,1),"",4)

44 POST ID: s X=cm K:\$L(X)>64!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

10 HOME PHONE NUMBER

FREE TEXT

K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 13

28 STORED: \$p(^ICC(DA,1),"",5)

44 POST ID: s X=cm K:\$L(X)>15!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

14.5 EMPLOYEE COST CENTER

FREE TEXT

K:\$L(X)>6!(\$L(X)<1) X

M/FORM FID # 96

28 STORED: \$p(^ICC(DA,2),"",4)

44 POST ID: s X=cm K:\$L(X)>6!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

11 AGENCY

FOINTER 1074' AGENCY(0,

M/FORM FID # 49

20 CONVERSION CODE: s

cm=\$s(cm="": "", '(Sd(^AGENCY(0,cm,0))#10):cm,1:\$p(^AGENCY(0,cm,0),"",2))

28 STORED: \$p(^ICC(DA,2),"",1)

11.5 AGENCY UNIT

FOINTER 1074' AGENCY(0,

M/FORM FID # 50

20 CONVERSION CODE: s

cm=\$s(cm="": "", '(Sd(^AGENCY(0,cm,0))#10):cm,1:\$p(^AGENCY(0,cm,0),"",2))

28 STORED: \$p(^ICC(DA,2),"",2)

12 SHOP

FREE TEXT

K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 16

28 STORED: \$p(^ICC(DA,2),"",3)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13 DUTY STATION FOINIER 1062' INJ(1062,

M/FORM FID # 98

20 CONVERSION CODE: s
 cm=\$s(cm="":",',(\$d(`INJ(1062,cm,0))#10):cm,1:\$p(`INJ(1062,cm,0),"",1))
 28 STORED: \$p(`IOC(DA,9),"",1)
 30 POSTFIELD CODE:
 S U""
 i \$d(new) s:new="" \$p(`IOC(DA,9),U,2,99)-\$p(`INJ(1062,new,0),U,2,99)

14 REPORTING OFFICE FOINIER 1063' INJ(1063,

M/FORM FID # 103

20 CONVERSION CODE: s
 cm=\$s(cm="":",',(\$d(`INJ(1063,cm,0))#10):cm,1:\$p(`INJ(1063,cm,0),"",1))
 28 STORED: \$p(`IOC(DA,10),"",1)
 30 POSTFIELD CODE:
 S U""
 i \$d(new) s:new="" \$p(`IOC(DA,10),U,2,10)-\$p(`INJ(1063,new,0),U,2,10)

WS EMPLOYMENT INFORMATION

DATA ELEMENT DATA TYPE FOINIER GLOBAL REFERENCE OR INPUT VERIFICATION

15 OCCUPATION FOINIER 1001' DIZ(1001,

M/FORM FID # 20

20 CONVERSION CODE: s
 cm=\$s(cm="":",',(\$d(`DIZ(1001,cm,0))#10):cm,1:\$p(`DIZ(1001,cm,0),"",2))
 28 STORED: \$p(`IOC(DA,3),"",1)
 32 POST FILING CODE: d setinck`infrm

16 TENURE

SET OF CODES

TEMP:TEMPORARY
 NON:NONCOMPETING
 CARE:CAREER
 CARC:CAREER-CONDITIONAL

M/FORM FID # 21

20 CONVERSION CODE:
 S cm=\$p(\$p("TEMP:TEMPORARY;NON:NONCOMPETING;CARE:CAREER;
 CARC:CAREER-CONDITIONAL;";",cm":",2);",1),cm=\$p(cm,"",1)
 28 STORED: \$p(`IOC(DA,3),"",3)

17 PAY PLAN

FREE TEXT

K:SL(X)>2!(SL(X)<2)!(X72A) X

M/FORM FID # 22

28 STORED: \$p(`IOC(DA,3),"",4)
 44 POST ID: s X=cm K:SL(X)>2!(SL(X)<2)!(X72A) X s:\$d(X) cm="" s:cm"" cm=""

18 PAY GRADE FREE TEXT K:\$L(X)>3!(\$L(X)<2) X

M/FORM FID # 23

28 STORED: \$p(^ICC(DA,3),"",5)

44 POST ID: s X=cm K:\$L(X)>3!(\$L(X)<2) X s:'\$d(X) cm="" s:cm[""" cm=""

19 PAY STEP FREE TEXT K:\$L(X)>2!(\$L(X)<1) X

M/FORM FID # 24

28 STORED: \$p(^ICC(DA,3),"",6)

44 POST ID: s X=cm K:\$L(X)>2!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

21 PAY RATE NUMERIC FORMAT S:X"\$" X=\$P(X,"\$",2) K:+X'=\$X(X'?.N1".2N)!(X>99999)!(X<0) X

M/FORM FID # 92

28 STORED: \$p(^ICC(DA,3),"",8)

20 PAY BASIS

SET OF CODES
H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-WEEKLY
A:ANNUALLY
D:DAILY

M/FORM FID # 93

20 CONVERSION CODE:

s cm=\$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;A:ANNUALLY;D:DAILY
;",";" cm ":"",2),";",1) ,cm=\$p(cm,";",1)

28 STORED: \$p(^ICC(DA,3),"",7)

22 SHIFT

SET OF CODES
1:DAY
2:SWING
3:GRAVEYARD

M/FORM FID # 28

20 CONVERSION CODE: s

cm=\$p(\$p("1:DAY;2:SWING;3:GRAVEYARD;",";" _ ":"",2),";",1) ,cm=\$p(cm,";",1)

28 STORED: \$p(^ICC(DA,3),"",9)

21.5 PAY STATUS

FOUNDER 1093' INU(1093,

M/FORM FID # 27

20 CONVERSION CODE: s

cm=\$s(cm="":"",('\$d(^INU(1093,cm,0))#10):cm,1:\$p(^INU(1093,cm,0),"",2))

28 STORED: \$p(^ICC(DA,3),"",10)

23 MONTHS ON JOB NUMERIC FORMAT K:+X'-X1(>999999999)!(X<0)!(X2.E1)".1N.N) X

M/FORM FID # 29

28 STORED: \$p(^ICC(DA,3),"",11)

24 FEGLI CODE FREE TEXT K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 30

28 STORED: \$p(^ICC(DA,3),"",12)

44 POST ID: s X=cm K:\$L(X)>15!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

25 FEHB CODE FREE TEXT K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 31

28 STORED: \$p(^ICC(DA,3),"",13)

44 POST ID: s X=cm K:\$L(X)>15!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

303 REGULAR WORK SCHEDULE FREE TEXT K:\$L(X)>15!(\$L(X)<3) X

M/FORM FID # 120

28 STORED: \$p(^ICC(DA,300),"",4)

44 POST s X=cm K:\$L(X)>15!(\$L(X)<3) X s:'\$d(X) cm="" s:cm[""" cm=""

302 REGULAR WORK HOURS FREE TEXT K:\$L(X)>14!(\$L(X)<6) X

M/FORM FID # 119

28 STORED: \$p(^ICC(DA,300),"",3)

44 POST ID: s X=cm K:\$L(X)>14!(\$L(X)<6) X s:'\$d(X) cm="" s:cm[""" cm=""

303.5 REGULAR NUMBER OF HOURS NUMERIC FORMAT K:+X'-X1(X>24)!(X<1)!(X2.E1)".1N.N) X

M/FORM FID # 118

28 STORED: \$p(^ICC(DA,300),"",6)

303.6 PARTTIME WORK HOURS PER NUMERIC FORMAT K:+X'-X1(X>8)!(X<1)!(X2.E1)".1N.N) X

M/FORM FID # 121

28 STORED: \$p(^ICC(DA,300),"",7)

402 SUPERVISOR'S NAME FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 122

28 STORED: \$p(^ICC(DA,402),"",1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

402.i SUPERVISOR'S TITLE

FREE TEXT

K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 123

28 STORED: Sp(ICC(DA,402),"",2)

29 PREFIELD CODE: S:obi(122)=" " skip="D"

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fmfm s U=""
i \$d(file),file=1,\$d(New),(New=0) d c600^T3CPMF

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0`fmfm

120 CAUSE OF INJURY POINTER 1026' INJ(1026,

M/FORM FID # 68

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1026,cm,0))#10):cm,1:\$p(`INJ(1026,cm,0),"",2))
28 STORED: \$p(`ICC(DA,110),"",5)

101 NATURE OF INJURY CODE POINTER 1017' INJ(1017,

M/FORM FID # 49

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1017,cm,0))#10):cm,1:\$p(`INJ(1017,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",2)
32 POST FILING CODE: d setindx`fmfm

100 BODY PART CODE POINTER 1018' INJ(1018,

M/FORM FID # 48

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1018,cm,0))#10):cm,1:\$p(`INJ(1018,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",1)

126 SOURCE OF INJURY POINTER 1027' INJ(1027,

M/FORM FID # 51

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1027,cm,0))#10):cm,1:\$p(`INJ(1027,cm,0),"",2))
28 STORED: \$p(`ICC(DA,120),"",3)
32 POST FILING CODE: d setindx`fmfm

26.5 SOURCE CODE FREE TEXT R:SL(X)>61(SL(X)<1) X

M/FORM FID # 50

28 STORED: \$p(`ICC(DA,120),"",9)
44 POST ID: s %cm R:SL(X)>61(SL(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

102 INJURY OSHA CODE POINTER 1057' INJ(1057,

M/FORM FID # 16

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1057,cm,0))#10):cm,1:\$p(`INJ(1057,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",3)

113 BLDG/SHIP FREE TEXT R:SL(X)>151(SL(X)<1) X

M/FORM FID # 64

28 STORED: \$p(^ICC(DA,100),"",14)

44 POST ID: s %cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

114 SPECIFIC LOCATION FREE TEXT K:SL(X)>40!(SL(X)<1) X

M/FORM FID # 65

28 STORED: \$p(^ICC(DA,100),"",15)

44 POST ID: s %cm K:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

INJURY LOG ENTRY Window 2

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

110 REPORTABLE CASE	SET OF CODES	Y:YES N:NO
---------------------	--------------	---------------

M/FORM FID # 18

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"_",1),cm=\$p(cm,"_",1)

28 STORED: \$p(^ICC(DA,100),"",11)

109 RECORDABLE CASE	SET OF CODES	Y:YES N:NO R:REMOVED
---------------------	--------------	----------------------------

M/FORM FID # 22

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;R:REMOVED;"_"_":",2),"_",1),cm=\$p(cm,"_",1)

28 STORED: \$p(^ICC(DA,100),"",10)

134 RECREATION INJURY	SET OF CODES	Y:YES N:NO
-----------------------	--------------	---------------

M/FORM FID # 23

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"_",1),cm=\$p(cm,"_",1)

28 STORED: \$p(^ICC(DA,13),"",3)

111 TERMINATION TRANSFER	SET OF CODES	Y:YES N:NO
--------------------------	--------------	---------------

M/FORM FID # 21

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"_",1),cm=\$p(cm,"_",1)

28 STORED: \$p(^ICC(DA,100),"",12)

104 LOST TIME ACCIDENT	SET OF CODES	Y:YES N:NO
------------------------	--------------	---------------

M/FORM FID # 19

105 LOST WORK DAYS NUMERIC FORMAT K:+X'-X!(X>999)!(X<0)!(X?.E1". "1N.N) X

106 LIGHT DUTY DAYS NUMERIC FORMAT K:+X'-X!(X>999)!(X<0)!(X?.E1". "1N.N) X

28 STORED: Sp(^IOC(DA,100),"",7)

```
20 CONVERSION CODE:  = cm=Sp(,":Y:YES;N:NO;"",":_cm_":",2),":",1),cm=Sp(cm,":",1)
28 STORED:  Sp(^ICC(DA,100),""",9)
```

```
20 CONVERSION CODE: S=Sp(Sp("Y: YES; N: NO; ", "_": "277; ", 1), am=Sp(am, ": ", 1)
28 STORED: Sp("ICC(DA, 12)", "", 4)
```

```
20 CONVERSION CODE: s=amSp(Sp("Y: YES; N: NO; ", "_am_":",2),",",1),amSp(am,"",1)
28 STORED: Sp("IOC(DA,13)", "",2)
```

```
20 CONVERSION CODE: %CONSP(%("Y: YES; N: NO; ", "%CON": ", 2), ", ", 1), CONSP(CON, ": ", 1)
28 STORED: %("IOC(DA, 13), "", 5)
```

20 CONVERSION CODE:
 S X= (" S: +Se (cm,4,5) X=X Sp ("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",

+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) " " Se(cm "000",11,12) s X=X " "
 s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,100),"",4)

127 AGENT OF ACCIDENT FOINIER 1068' INJ(1068,

M/FORM FID # 28

20 CONVERSION CODE: s cm=Se(cm="":",',(\$d(^INJ(1068,cm,0))#10):cm,1:Sp(^INJ(1068,cm,0),"",2))
 28 STORED: Sp(^ICC(DA,12),"",4)

128 AGT OF ACC PARTICULAR FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 31

28 STORED: Sp(^ICC(DA,12),"",5)
 44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

129 UNSAFE PRACTICE FREE TEXT K:SL(X)>40!(SL(X)<1) X

M/FORM FID # 32

28 STORED: Sp(^ICC(DA,12),"",6)
 44 POST ID: s X=cm K:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

130 UNSAFE CONDITION FREE TEXT K:SL(X)>40!(SL(X)<1) X

M/FORM FID # 33

28 STORED: Sp(^ICC(DA,12),"",7)
 44 POST ID: s X=cm K:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

132 STRESSOR INVOLVED INJURY FOINIER 1083' STRESS(0,

M/FORM FID # 34

20 CONVERSION CODE: s cm=Se(cm="":",',(\$d(^STRESS(0,cm,0))#10):cm,1:Sp(^STRESS(0,cm,0),"",1))
 28 STORED: Sp(^ICC(DA,13),"",1)

131 PPE INVOLVED INJURY FREE TEXT K:SL(X)>40!(SL(X)<1) X

M/FORM FID # 35

28 STORED: Sp(^ICC(DA,12),"",8)
 44 POST ID: s X=cm K:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
 +s filling=Se('\$d(filling):lvl,lvl>filling:lvl,1:filling)
 +d upcntr^fnfcm S U=""

4.3.2 Enter Claim Forms [T3C CLAIM].

Purpose of transaction: This option is used to enter a claims form. Form #726 is called which selects the next form to be entered based on the last form filed.

Transaction Description: This transaction is called from a Menu option [T3C CLAIM] in the OSHR&S application package.

Type of transaction: Action which uses M/FORM #726 as a selection routine to enter Claims forms. Figures 4-8 and 4-9 show the forms that will be selected and the criterion for each.

Global Dictionary Reference: M/FORM— ^ptg(903,726,

Type of Menu Option: Action

Action/Routine Code: s iform=726,clear=1 k wipe,DIC(0) d ^ptrform

Output: Stored in Injury File #1005, reference global: ^ICC(N, and in Employee File #1004.2, reference global: ^Employ(DA,"ICC",F, (N=internal case number, DA= internal employee index, F= internal form number).

Retention: Once entered a case may not be deleted without programmer access. Claim forms can be entered and edited in the Injury file, but once they have been filed into the employee's permanent record, only a programmer can make a change to the record.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: Created using M/FORM-FileMan Interface, should not change form.

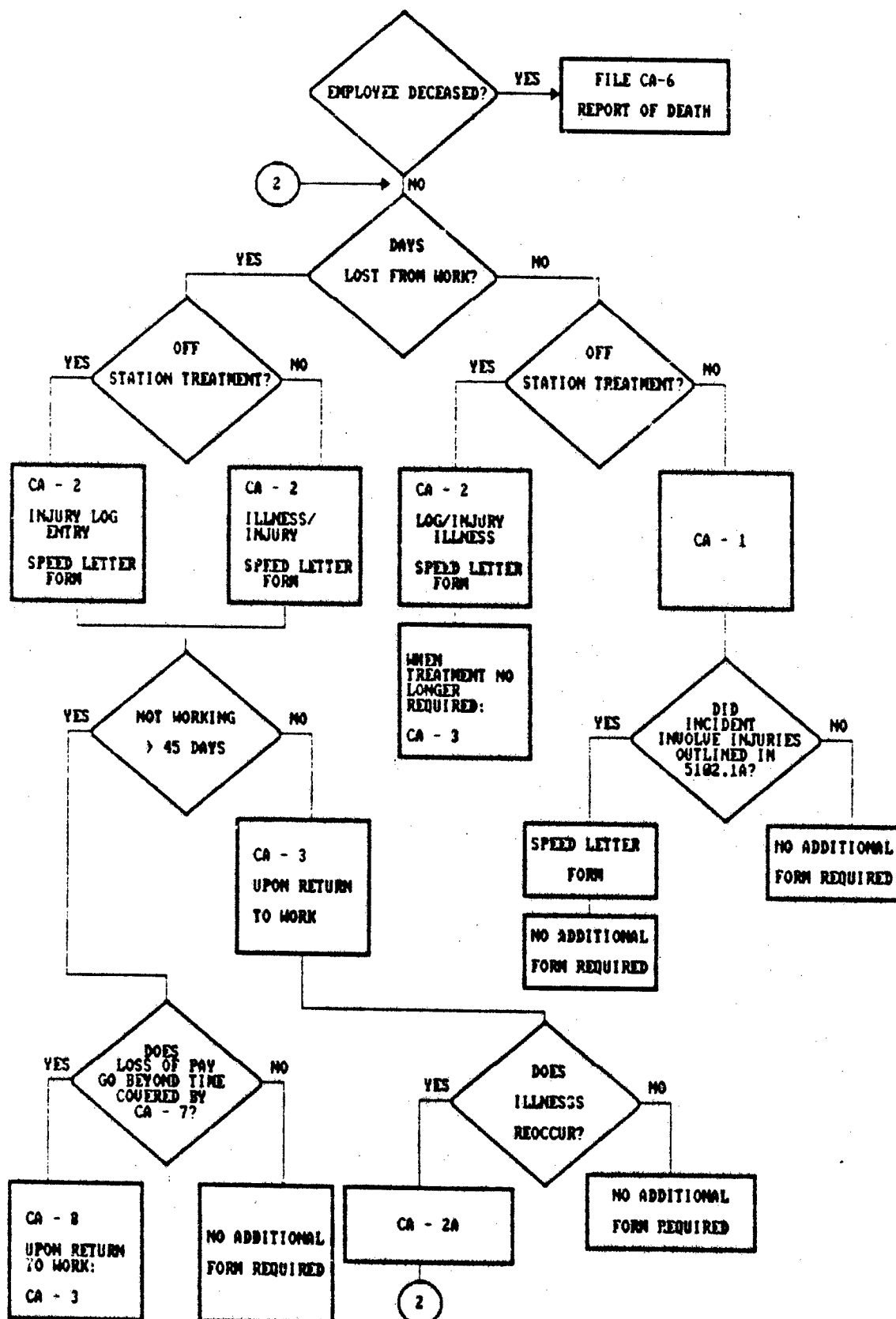
Error Recovery: Error trap by Kernel ^%ER routine

Report Format Requirements: None

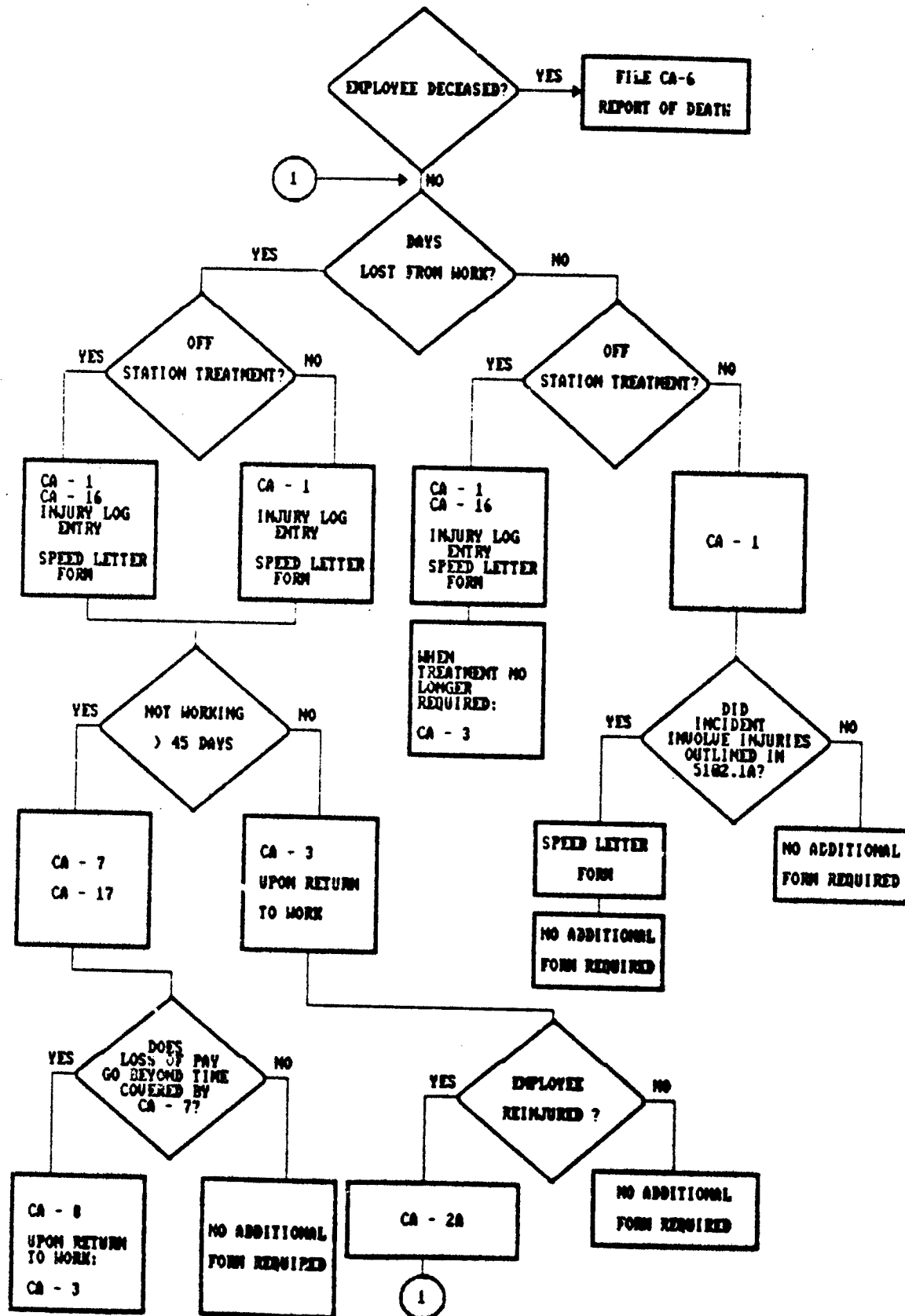
Inputs: See Appendix A for claim form input templates and M/FORM documentation. The inputs for form #726 are found below with the M/FORM documentat. on the following two pages.

ENTER CLAIM FORM	
NAME _____	INJURY/ILLNESS DATE _____

ILLNESS



INJURY



M/FRM: ENTER CLAIM FORM (\$726)

I. PRE-FORM CODE

```
n file,rptfile,New,filing,DA s:'$d(nocomp) file='ptc(iform,99)
+s:$d(nocomp) file='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),DT="" d ~DT s DT=Y k X,Y
d sec~mfmm
```

II. WINDOWS

ENTER CLAIM FORM Window 1

ADA ELEMENT	DATA TYPE	FOINIER GLOBAL REFERENCE OR INPUT VERIFICATION
-------------	-----------	--

2 NAME FOINIER 1004' EMPLOY(

M/FRM FID # 1

20 CONVERSION CODE: s cm=\$s(cm=""', '\$d('EMPLOY(cm,0))#10):cm,1:\$p('EMPLOY(cm,0)',"",1))

28 STORED: \$p('IOC(DA,0)',"",3)

30 POSTFIELD CODE:

I New=1 s cbi(1)="" msg="Claim not logged!",y=24

S:\$D(cbi(1)) NAME=\$P(cbi(1),cb,1),NAMEPIR=\$P(cbi(1),cb,2)

32 POST FILING CODE: d setindx~mfmm

44 POST ID: i New=1 s msg="Claim is not logged!",y=24,cm=""

54 INDEX: 'IOC("C")

55 LOOKUP NDE:

1~30~\$p('IOC(lids,0)',"",3)

5~23~\$p('IOC(lids,0)',"",2)

1 INJURY/ILLNESS DATE & TI DATE S ~DT="ET" D ~DT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FRM FID # 5

20 CONVERSION CODE:

s X="(" s:+\$s(cm,4,5) X=X \$p('JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC',".")

,+\$s(cm,4,5)) " " s:+\$s(cm,6,7) X=X \$s(cm,6,7) " " s X=X (1700+\$s(cm,1,3))

s:\$p(cm,".",2)'="" X=X " at " \$s(cm~"000",9,10) " " \$s(cm~"000",11,12) s X=X " " s:X="(1700)" X="" s cm=X

28 STORED: \$p('IOC(DA,0)',"",2)

29 PREFIELD CODE: i New=1!(cbi(1)=""') s msg="Claim not logged!",y=24 d hprt~ptrslot2

s cm=""',endform="R"

32 POST FILING CODE: d setindx~mfmm

44 POST ID: i New=1!(cbi(1)=""') s msg="Claim not logged!",y=24 d hprt~ptrslot2

s cm=""',endform="R"

54 INDEX: 'IOC("D")

55 LOOKUP NDE:

5~23~\$p('IOC(lids,0)',"",2)

1~30~\$p('IOC(lids,0)',"",3)

208 CLAIM FORM LAST RECEIVED POINTER 1002' FORMS(

M/FORM FID # 6

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(`FORMS(cm,0))#10):cm,1:\$p(`FORMS(cm,0),"",1))

28 STORED: \$p(`ICC(DA,200),"",9)

209 CLAIM FORM NEXT POINTER 1002' FORMS(

M/FORM FID # 4

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(`FORMS(cm,0))#10):cm,1:\$p(`FORMS(cm,0),"",1))

28 STORED: \$p(`ICC(DA,200),"",10)

29 PREFIELD CODE:

S F="" i \$d(DA),DA="" s:\$d(`ICC(DA,200)) F=\$P(`ICC(DA,200),U,10) K:F="" F

I \$D(F) K:\$D(`FORMS(F,0)) F I \$D(F) S F=\$P(`FORMS(F,0),"",5) K:F="" F

I \$D(F) K:\$D(`ptg(903,F)) F

I \$D(F) S iform=F,clear=1,ERFLAG=1,New=1 k F g `ptrform

31 BRANCHING CODE: i \$d(FORM) i FORM="" s newform=FORM,clear=1,New=1 g do`ptrform

44 POST ID: I cm="" d `T3CFK

III. POST-FORM CODE

I \$D(FILES),(\$D(DA)) S \$P(`ICC(DA,200),U,10)="" K FILES,ERFLAG

i \$d(file),file,New=0 s:\$d(lvl) lvl=0

+s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)

+d upcntr`fufm

4.3.3 Continuation of Pay Enter/Edit [T3C COP].

Purpose of transaction: Using an M/FORM input template, this option allows the claims worker to enter/edit continuation of pay information directly from the claim forms or other sources.

Transaction Description: This transaction is called from a Menu option [T3C COP] in the OSHRKS application package.

Type of transaction: Input template which uses M/FORM #730 to enter continuation of pay information directly in the COP file #1008.

Global Dictionary Reference: M/FORM— `ptg(903,730,

Type of Menu Option: Action

Action/Routine Code: s ifon=730,clear=1 k wipe,DIC(0) d `ptrform

Output: Stored Compensation File #1008, reference global: `INJ(1008,DA and in Injury File #1005, reference global: `ICC(N (DA= internal employee index, N= internal case number)

Retention: M/FORM file should not be changed. Data from the COP File #1008 can be archived periodically.

Security/Privacy: Calls `T2USER to verify agency access.

Flexibility: Created using M/FORM-FileMan Interface, should not change form.

Error Recovery: Error trap by Kernel `IER routine

Report Format Requirements: None

Inputs: See below.

CONTINUATION OF PAY	
NAME	
INJURY/ILLNESS DATE & TIME	
CONTINUATION OF PAY BEGIN DATE	
CONTINUATION OF PAY NOT TO EXCEED DATE	
RETURN TO WORK DATE	RETURN TO WORK HOURS
RECURRENCE FLAG	CONTINUATION OF PAY DAYS USED
JOB ORDER COST	WORK HOURS
IS THE CLAIM CONTROVERTED?	
CONTROVERSION SUSTAINED BY OWCP?	

M/FORM: CONTINUATION OF BAY (\$730)

I. PRE-FORM CODE

```
n filen,rptfile,new,filing,DA s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($i),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm
```

II. WINDOWS

CONTINUATION OF BAY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	POINTER 1004'	EMPLOY(

M/FORM FID # 1

```
20 CONVERSION CODE: s cm=$s(cm=""',('$d(EMPLOY(cm,0))%10):cm,1:$p(EMPLOY(cm,0),"",1))
28 STORED: $p(ICC(DA,0),"",3)
32 POST FILING CODE: d setindx`fmfm
44 POST ID: I $D(new) s:'$d(ICC("C",new,x1)) cm="",msg="Claim not logged!",y=24
54 INDEX: ICC("C")
55 LOOKUP NODE:
1^30`$p(ICC(lids,0),"",3)
2^23`$p(ICC(lids,0),"",2)
```

1 INJURY/ILLNESS DATE & TI DATE S %DT="ET" D %DT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 2

```
20 CONVERSION CODE:
s X="(" s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at "$e(cm"000",9,10]' ":" $e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: $p(ICC(DA,0),"",2)
32 POST FILING CODE: d setindx`fmfm
54 INDEX: ICC("D")
55 LOOKUP NODE:
2^23`$p(ICC(lids,0),"",2)
1^30`$p(ICC(lids,0),"",3)
```

210 COP BEGIN DATE DATE S %DT="ET" D %DT S X=Y K:Y<1 X

M/FORM FID # 3

```
20 CONVERSION CODE:
s X="(" s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at "$e(cm"000",9,10]' ":" $e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X
```

28 STORED: \$p(^ICC(DA,220),"",1)

210.1 COP NOT TO EXCEED DATE DATE S %DT="E" D ^%DT S %Y K:Y<1 X

M/FORM FID # 4

20 CONVERSION CODE:

s %="(" s:+\$e(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,"",2)'="" %K " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s %K_"")
s:\$(1700) %="" s cmK

28 STORED: \$p(^ICC(DA,220),"",2)

210.3 COP RETURN TO WORK DATE DATE S %DT="E" D ^%DT S %Y K:Y<1 X

M/FORM FID # 5

20 CONVERSION CODE:

s %="(" s:+\$e(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,"",2)'="" %K " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s %K_"")
s:\$(1700) %="" s cmK

28 STORED: \$p(^ICC(DA,220),"",4)

210.4 COP RETURN TO WORK HOURS NUMERIC FORMAT K:+X'~K1(X>24)!(X<0)!(X?.E1".IN.N) X

M/FORM FID # 6

28 STORED: \$p(^ICC(DA,220),"",5)

210.5 COP RECURRENCE FLAG SET OF CODES Y:YES
N:NO

M/FORM FID # 7

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",",",2),",",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,220),"",6)

210.6 COP DAYS USED NUMERIC FORMAT K:+X'~K1(X>9999)!(X<0)!(X?.E1".IN.N) X

M/FORM FID # 8

28 STORED: \$p(^ICC(DA,220),"",7)

210.7 COP JOB ORDER COST NUMERIC FORMAT S:X1"\$" %=\$P(X,"\$",2) K:+X'~K(X'?.N1".2N)!(X>999999.99)
!(X<0) X

M/FORM FID # 9

28 STORED: \$p(^ICC(DA,220),"",8)

210.2 COP HOURS OF DAY NUMERIC FORMAT R: +X' =X!(X>24):(X<0):(X?.E1". "1N.N) X

M/FORM FID # 13

28 STORED: \$p(^ICC(DA,220),"^",3)

205 CLAIM CONTROVERSED	SET OF CODES	Y:YES N:NO
------------------------	--------------	---------------

M/FORM FID # 11

```
20 CONVERSION CODE:  s cm Sp( Sp( " ; Y: YES; N: NO; " ; " _cm _": " , 2) , " ; " , 1) , cm Sp( cm , " : " , 1)
28 STORED:  Sp( ^ICC(DA, 200) , " " , 6)
```

205.2 CONTROL SUSTAINED BY CMC SET OF CODES Y:YES
N:NO

M/FORM FTD # 12

20 CONVERSION CODE: S:cm\$P(\$P(;"Y:YES;N:NO;"_"_cm_:"_",2),"_",1),cm\$P(cm,"_",1)
28 STORED: \$P("ICC(DA,200),"_",1)
29 PREFIELD CODE: S:'\$D(chl(11)) skip="D" i '\$d(skip) s:chl(11)/["Y" skip="D"

III. POST-FORM CODE

```

    $d(file),file,New=0 s:'$d(lvl) lvl=0
+ s:filling:~s('$(filling):lvl,lvl>filling:lvl,1:filling)
    I $d(file),file=1 d ~T30CP
    +d upcntr~sumfm

```

4.3.4 Compensation Pay Enter/Edit [T3COMP]

Purpose of transaction: Using an M/FORM input template, this option allows the claims worker to enter/edit compensation pay information directly from the claim forms or other sources.

Transaction Description: This transaction is called from a Menu option [T3C COMP] in the OSHRKS application package.

Type of transaction: Input template which uses M/FORM #731 to enter continuation of pay information directly in the COP file #1009.

Global Dictionary Reference: M/FORM-- ^ptg(903,731,

Type of Menu Option: Action

Action/Routine Code: s iform=731,clear=1 k wipe,DIC(0) d ^ptrform

Output: Stored Compensation File #1009, reference global: ^INJ(1009,DA and in Injury File #1005, reference global: ^ICC(N (DA= internal employee index, N= internal case number)

Retention: M/FORM file should not be changed. Data from the Compensation file #1009 can be archived periodically.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: Created using M/FORM-FileMan Interface, should not change form.

Error Recovery: Error trap by Kernel ^%ER routine

Report Format Requirements: None

Inputs: See below.

_____COMPENSATION PAY_____	
NAME _____	
INJURY/ILLNESS DATE & TIME _____	
COMPENSATION TYPE _____	COMP RECURRENCE? _____
DATE COMPENSATION BEGAN _____	COMP BEGIN HOURS _____
DATE COMPENSATION ENDED _____	COMP END HOURS _____
DATE COMPENSATION RECEIVED _____	AMOUNT OF COMPENSATION \$ _____
CLAIM CONTROVERTED? _____	CONTROVERSION SUSTAINED BY OWCP? _____
COMP LEAVE STATUS _____	
JOB TRANSFER FOR MEDICAL REASONS? _____	

M/FORM: COMPENSATION PAY (#731)

I. PRE-FORM CODE

n filen,rptfile,new,filing,DA s:'\$d(nocomp) filen='ptc(ifcom,99)
+s:\$d(nocomp) filen='ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm

II. WINDOWS

COMPENSATION PAY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	POINTER 1004'	EMPLOY(

M/FORM FID # 1

20 CONVERSION CODE: s cm=\$e(cm=""',('(\$d`EMPLOY(cm,0))#10):cm,1:\$p`EMPLOY(cm,0),"",1))
28 STORED: \$p`IOC(DA,0),"",3)
32 POST FILING CODE: d setindx`fmfm
44 POST ID: I SD(new) s:'\$d`IOC("C",new,x1)) cm="" ,msg="Claim not logged!",y=24
54 INDEX: `IOC("C")
55 LOOKUP NODE:
1`20`\$p`IOC(Lids,0),"",3)
2`23`\$p`IOC(Lids,0),"",2)

1 INJURY/ILLNESS DATE & TI DATE S %DT="ET" D %DT S %Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 2

20 CONVERSION CODE:
s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)`="" X=X " at " \$e(cm`000",9,10)`:" \$e(cm`000",11,12) s X=X`")
s:X="(1700)" X="" s cm=X
28 STORED: \$p`IOC(DA,0),"",2)
32 POST FILING CODE: d setindx`fmfm
54 INDEX: `IOC("D")
55 LOOKUP NODE:
2`23`\$p`IOC(Lids,0),"",2)
1`20`\$p`IOC(Lids,0),"",3)

256.3 COMPENSATION TYPE	SET OF CODES	C:COMPENSATION S:SCHEDULED AWARD
-------------------------	--------------	-------------------------------------

M/FORM FID # 3

20 CONVERSION CODE: s cm=\$p(\$p("C:COMPENSATION;S:SCHEDULED AWARD;",";","_cm_":",2)",";"),
cm=\$p(cm,".",1)

28 STORED: \$p(^ICC(DA,251),"",4)

256.8 COMP RECURRENCE FLAG SET OF CODES Y:YES
N:NO

M/FORM FID # 14

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";","_cm_":",2)",";","1),cm=\$p(cm,";",1)
28 STORED: \$p(^ICC(DA,251),"",9)

256 COMP REC'D FROM DATE S %DT="E" D %DT S %Y K:Y<1 X

M/FORM FID # 4

20 CONVERSION CODE:
s %="(" s:+\$e(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" " %K " at " \$e(cm^"000",9,10) ":" \$e(cm^"000",11,12) s %K_"")
s:%="(1700)" %=" " s cm=K

28 STORED: \$p(^ICC(DA,251),"",1)

256.4 COMP BEGIN HOURS NUMERIC FORMAT K:+X'-X'(X>24)!(X<0)!(X2.E1". "IN.N) X

M/FORM FID # 6

28 STORED: \$p(^ICC(DA,251),"",5)

256.1 COMP REC'D THRU DATE S %DT="E" D %DT S %Y K:Y<1 X

M/FORM FID # 5

20 CONVERSION CODE:
s %="(" s:+\$e(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" " %K " at " \$e(cm^"000",9,10) ":" \$e(cm^"000",11,12) s %K_"")
s:%="(1700)" %=" " s cm=K

28 STORED: \$p(^ICC(DA,251),"",2)

256.5 COMP END HOURS NUMERIC FORMAT K:+X'-X'(X>24)!(X<0)!(X2.E1". "IN.N) X

M/FORM FID # 7

28 STORED: \$p(^ICC(DA,251),"",6)

256.7 COMP REC'D DATE DATE S %DT="E" D %DT S %Y K:Y<1 X

M/FORM FID # 10

20 CONVERSION CODE:
s %="(" s:+\$e(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" " %K " at " \$e(cm^"000",9,10) ":" \$e(cm^"000",11,12) s %K_"")
s:%="(1700)" %=" " s cm=K

28 STORED: \$p(^ICC(DA,251),"",8)

56.2 COMP RECD GROSS NUMERIC FORMAT S:X("S" X=\$p(X,"S",2) K:+X=\$E(X'7.N1"."2N)!(X0999999.99)
I(X<0) X

M/FORM FID # 9

28 STORED: \$p(^ICC(DA,251),"",3)

205 CLAIM CONFOVERIED SET OF CODES Y:YES
N:NO

M/FORM FID # 11

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,200),"",6)

205.2 CONIROV SUSTAINED BY OAC SET OF CODES Y:YES
N:NO

M/FORM FID # 12

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,200),"",11)
29 PREFIELD CODE: s:'\$d(cbi(11)) skip="D" i '\$d(skip) s:cbi(11)'["Y" skip="D"]

256.6 COMP LEAVE STATUS SET OF CODES S:SICK LEAVE
P:ADMINISTRATIVE
L:LOAF
T:TEMP OFF STATION
A:ANNUAL LEAVE (VACATION)

M/FORM FID # 8

20 CONVERSION CODE: s cm=\$p(\$p("S:SICK LEAVE;P:ADMINISTRATIVE;L:LOAF;T:TEMP OFF STATION;
A:ANNUAL LEAVE (VACATION)",";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,251),"",7)

138 JOB TRANSFER MED REASONS SET OF CODES Y:YES
N:NO

M/FORM FID # 13

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,130),"",7)

III. POST-FORM CODE

I \$d(file) d:file=1 ^T3COMP
i \$d(file),file,new=0 s:'\$d(lvl) lvl=0
+s filing=\$d('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fmfm

4.3.5 Print Claim Forms [T3CPRNT].

Purpose of transaction: After claim forms have been entered and permanently stored in the Employee file, this option allow the user to print claim form facsimiles.

Transaction Description: This transaction is called from a Menu option [T3CPRNT] in the OSHRKS application package.

Type of transaction: Routine Output

Global Dictionary Reference: VA FileMan ^DD(1002, ^FORMS(^EMPLOY(DA,"ICC",F,

Type of Menu Option: Routine

Action/Routine Code: T3CF

Inputs: CA form number, Employee Name

Output: See Appendix B for Facsimile outputs

Retention: Data from the Claims forms is stored in the Employee File #1004.2 and to be retained for historical purposes and should be archived periodically.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: Form facsimiles can be changed through the VA File Manager enter/edit option for file #1002.

Error Recovery: Error trap by Kernel ^%ER routine

Report Format Requirements: See appendix B for requirements.

4.3.6 Claims Pay-Receive Reports [T3C CLAIMS PAY DATA]

Purpose of transaction: Three payment output templates have been created to print reports; all payment information, continuation of pay, and compensation pay.

Transaction Description: This transaction is called from a Menu option [T3C CLAIMS PAY DATA] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference:

Type of Menu Option: Menu

Action/Routine Code:

Inputs: Transaction allows person to select from three output templates; [T3CPAY1 PAY TYPES] Pay Information by Employee, [T3CPAY4] Continuation of Pay, and [T3CPAY3 COMP DATA] Compensation Data.

Output:

Retention: This option was created as an example of the type of payment information reports that the site manager can create. Options can be added or deleted at the discretion of the site manager.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^%ER routine

4.3.6.1 Pay Information by Employee [T3CPAY1 PAY TYPES]

Purpose of transaction: An output template using the VA FileMan print capabilities to show the regular, leave, COP, and compensation pay information by date.

Transaction Description: This transaction is called from a Menu option [T3CPAY1 PAY TYPES] in the OSHRKS application package. Requires 132 column device.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [PAY INFORMATION]

Output: PRINT OUT SAMPLE

PAY INFORMATION BY TYPE OF PAY RECEIVED

Name: _____	Case No: _____	Injury Date: MM/DD/YY _____			
Occupation: _____	Nature of Injury: _____				
	Begin Date	End Date	Rate	Basis	Gross Amount Received
CONTINUATION OF PAY MM/DD/YY MM/DD/YY					
SICK LEAVE PAY	MM/DD/YY	MM/DD/YY	_____	_____	_____
ANNUAL LEAVE PAY	MM/DD/YY	MM/DD/YY	_____	_____	_____

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^KER routine

Report Format Requirements:

PRINT TEMPLATE: [PAY INFORMATION]

HEADING: PAY INFORMATION BY TYPE OF PAY RECEIVED

READ ACCESS:

WRITE ACCESS:

<u>PRINT</u>	<u>FORMAT</u>
1. "Name:"	S1
2. NAME	X;L30
3. "Case No:"	C50
4. CASE SERIAL NUMBER	X;L8
5. "Injury Date:"	C80
6. NUMDATE(INJURY/ILLNESS DATE & TIME)	X
7. "Occupation:"	C8
8. OCCUPATION:OCCUPATION TITLE	X
9. "Nature of Injury:"	C80
10. NATURE OF INJURY CODE:NATURE OF INJURY	X
11. "Begin Date"	S1;C35
12. "End Date"	C50
13. "Rate"	C65
14. "Basis"	C80
15. "Gross Amount Received"	C105
16. DUP("-",91)	X;C35
17. PAY INFORMATION (MULTIPLE)	
A. TYPE OF PAY RECEIVED	X;C1
B. NUMDATE(DATE IMPLEMENTED)	X;C35
C. NUMDATE(END DATE)	X;C50
D. RATE	X;C65;L9;D2
E. BASIS	X;C80
F. GROSS AMOUNT RECEIVED	X;C105;L9;D2

4.3.6.2 Continuation of Pay [T3CPAY4]

Purpose of transaction: An output template using the VA FileMan print capabilities to show the COP status of individual cases and events.

Transaction Description: This transaction is called from a Menu option [T3CPAY4] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: INJ(1008,

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [CONTINUATION OF PAY]

Output: PRINT OUT SAMPLE

CONTINUATION OF PAY DATA

MM/DD/YY TIME PAGE #

Case No: _____ Name: _____ Injury Date: _____

Begin Date: MM/DD/YY Not To Exceed: MM/DD/YY
Recurrence? _____

Hours/Day COP Paid: _____

Days of COP Used: _____

Return To Work Date: MM/DD/YY

Hours/Day COP Paid on Return To Work: _____

Job Order Cost: _____

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel %ER routine

Report Format Requirements:

OUTPUT FROM WHAT FILE: COP DATA #1008

PRINT TEMPLATE: [CONTINUATION OF PAY]

HEADING: CONTINUATION OF PAY DATA

READ ACCESS:

WRITE ACCESS:

PRINT

1. "Case No."
2. CASE SERIAL NUMBER
3. "Name:"
4. NAME
5. "Injury Date:"
6. NUMDATE(INJURY/ILLNESS DATE & TIME)
7. "Begin Date:"
8. NUMDATE(COP BEGIN DATE)
9. "Not To Exceed:"
10. NUMDATE(COP NOT TO EXCEED DATE)
11. "Recurrence?"
12. COP RECURRENCE FLAG
13. DUP(" ",44)
14. "Hours/Day COP Paid:"
15. COP HOURS OF DAY
16. "Days of COP Used:"
17. COP DAYS USED
18. "Return To Work Date:"
19. NUMDATE(COP RETURN TO WORK DATE)
20. "Hours/Day COP Paid on Return To Work:"
21. COP RETURN TO WORK HOURS
22. DUP(" ",44)
23. "Job Order Cost:"
24. COP JOB ORDER COST

FORMAT

S1
X;L6
C20
X;L30
C58
X
S1;C5
X
C30
X
C58
X
X;C10
C10
X;L2
C10
X;L4
C10
X
C10
X;L2
X;C10
C5
X;L9;D2

4.3.6.3 Compensation Data [T3CPAY3 COMP DATA]

Purpose of transaction: An output template using the VA FileMan print capabilities to show the compensation status of individual cases and events.

Transaction Description: This transaction is called from a Menu option [T3CPAY3 COMP DATA] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: INJ(1009

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [COMPENSATION]

Output: PRINT OUT SAMPLE

COMPENSATION DATA

MM/DD/YY TIME PAGE #

Case No: _____ Name: _____ Injury Date: _____
Begin Date: MM/DD/YY End Date: MM/DD/YY Recurrence? _____

Compensation Type: _____
Leave Status: _____
Begin/End Hours: _____
Received Payment On: _____
Gross Amount Received: _____

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel %ER routine

Report Format Requirements:

PRINT TEMPLATE: [COMPENSATION]
HEADING: COMPENSATION DATA
READ ACCESS:
WRITE ACCESS:

PRINT

FORMAT

1. "Case No:"	S1
2. CASE SERIAL NUMBER	X;L8
3. "Name:"	C20
4. NAME	X;L30
5. "Injury Date:"	C58
6. NUMDATE(INJURY/ILLNESS DATE & TIME)	X
7. COMP INFORMATION (MULTIPLE)	
8. "Begin Date:"	C5
9. NUMDATE(COMP RECVD FROM)	X
10. "End Date:"	C30
11. NUMDATE(COMP RECVD THRU)	X
12. "Recurrence?"	C58
13. COMP RECURRENCE FLAG	X
14. DUP("=",39)	X;C10
15. "Compensation Type:"	C10
16. COMPENSATION TYPE	X
17. "Leave Status:"	C10
18. COMP LEAVE STATUS	X
19. "Begin/End Hours:"	C10
20. COMP BEGIN HOURS	X
21. "Received Payment On:"	C10
22. NUMDATE(COMP RECVD DATE)	X
23. "Gross Amount Received:"	C10
24. COMP RECVD GROSS	X;L9;D2
25. DUP("=",39)	X;C10

4.3.7 Tickler File Print/Display [T3CSR8 TICKLER]

Purpose of transaction: Print/Display Case Summary Report by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR8 TICKLER] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [TICKLER REPORT2]

Output: PRINT OUT SAMPLE

TICKLER REPORT

MM/DD/YY TIME PAGE #

CSN: _____	SHOP: _____
NAME: _____	SSN: _____
DATE OF INJURY: MM/DD/YY	
CURRENT STATUS: _____	DATE COP BEGAN: MM/DD/YY
TOTAL LWDays: _____	DATE RETURNED TO WORK: _____

TOTAL
COUNT

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel TICKLER routine

Report Format Requirements:

PRINT TEMPLATE: [TICKLER REPORT2]
HEADING: TICKLER REPORT
READ ACCESS:
WRITE ACCESS:

PRINT

1. "CSN:"
2. CASE SERIAL NUMBER
3. "SHOP:"
4. AGENCY UNIT
5. "NAME:"
6. NAME
7. "SSN:"
8. SSN
9. "DATE OF INJURY:"
10. INJURY/ILLNESS DATE & TIME
11. "CURRENT STATUS:"
12. "DATE COP BEGAN:"
13. COP BEGIN DATE
14. "TOTAL LWDays:"
15. LOST WORK DAYS
16. "DATE RETURNED TO WORK:"
17. D/T RETURNED TO WORK

FORMAT

C65
X (SUPPRESS HEADER)
C1
X!
S1;C5
X;L30
C35
X
C35
X
C12
C35
X
C13
X&
C35
X

4.3.8 Adjudication Status Report [T3CSR9 ADJUDICATION STATUS]

Purpose of transaction: An output template using the VA FileMan print capabilities to show the adjudication status of individual cases and events.

Transaction Description: This transaction is called from a Menu option [T3CSR9 ADJUDICATION STATUS] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [ADJUDICATION STATUS]

Output: PRINT OUT SAMPLE

ADJUDICATION STATUS REPORT

MM/DD/YY TIME PAGE #

NAME: _____
SSN: _____
CASE NO: _____ NOI: _____
DOI: _____

ADJ STATUS	DATE ENTERED	DATE REMOVED
23 NEED INFORMATION - FROM CLAIMANT & MEDIC	MM/DD/YY	MM/DD/YY
22 NEED INFORMATION - FROM MEDICAL SOURCE	MM/DD/YY	MM/DD/YY
24 NEED INFORMATION - FROM EMPLOYER	MM/DD/YY	MM/DD/YY
A03 CLAIM ACCEPTED BY OWCP - COP AUTHORIZED	MM/DD/YY	(PENDING)

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^IER routine

Report Format Requirements:

PRINT TEMPLATE: [ADJUDICATION STATUS]
HEADING: ADJUDICATION STATUS REPORT
READ ACCESS: @
WRITE ACCESS: @

PRINT

FORMAT

1. "NAME:"	S1
2. NAME	X (SUPPRESS HEADER)
3. "SSN:"	C50
4. SSN	X
5. "CASE NO:"	C3
6. CASE SERIAL NUMBER	X;L8
7. "NCI:"	C21
8. NATURE OF INJURY CODE:NATURE OF INJURY	X
9. "DOI:"	C60
10. NUMDATE(INJURY/ILLNESS DATE & ILLNESS)	X
11. "ADJ STATUS"	S1
12. "DATE ENTERED"	C50
13. "DATE REMOVED"	C65
14. DUP("-",80)	X;C1
15. ADJUDICATION STATUS (MULTIPLE)	
A. ADJUDICATION STATUS	X;C1
B. ADJUDICATION STATUS:STATUS DESCRIPTION	X;C5;L40
C. NUMDATE(DATE ENTERED)	X;C50
D. NUMDATE(DATE REMOVED)	X;C65
16. DUP("-",80)	X;C1

4.4 Safety Office Functions [T3CS SAFETY]. Menus under the Safety functions are as follows:

4.4.1 Enter/Edit Injury [T3CSLG]

Purpose of transaction: This menu is used to enter, edit, and update existing injury/illness data. This option used an M/FORM template for data entry.

Transaction Description: Menu option initialized when OSHRKS system is installed.

Type of transaction: Input

Global Dictionary Reference: M/FORM ^ptg(903,601

Type of Menu Option: Action

Action/Routine Code: S iform=601,clear=1 K DIC(0),wipe D ^ptrform

Output: Data stored in Injury file #1005 ^ICC(

Retention: Required for historical purposes should be archived.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: M/FORM input templates should be changed by programmer familiar with the VA FileMan -- M/FORM interface problems.

Error Recovery: Error trap by Kernel ^3ER routine

Report Format Requirements:

Inputs: See following pages for input templates and documentation.

INJURY LOG ENTRY

INJURY/ILLNESS DATE & TIME _____ CASE NUMBER _____
NAME _____

INJURY LOG ENTRY DATE _____
SHOP _____ SHIFT _____ OCCUPATION CODE _____
OCCUPATION _____
DUTY STATION _____ MONTHS ON JOB _____
PRIMARY SUPERVISOR _____ EMPLOYEE COST CENTER _____
SECONDARY SUPERVISOR _____ NONFATAL LOST WORKDAYS _____
WAS SUPERVISOR ACCIDENT REPORT FILED? _____
TYPE OF ACCIDENT _____

Press <Return> for WITNESS INFORMATION

CAUSE OF INJURY _____
NATURE OF INJURY _____
BODY PART _____
SOURCE OF INJURY _____ SOURCE CODE _____
INJURY OSHA CODE _____
BLDG/SHIP _____
SPECIFIC LOCATION _____

WITNESS INFORMATION

WITNESS NAME _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

Press <Return> for WITNESS STATEMENT

REPORTABLE CASE _____ RECORDABLE CASE _____ RECREATION INJURY _____
TERMINATION TRANSFER _____ LOST TIME ACCIDENT _____ LOST WORK DAYS _____
LIGHT DUTY DAYS _____ FIRST AID CASE _____ VEHICLE INJURY _____ TDY INJURY _____
TWO WHEEL VEHICLE INJURY _____ DATE OF DEATH _____
AGENT OF ACCIDENT _____
AGT OF ACC PARTICULAR _____
UNSAFE PRACTICE _____
UNSAFE CONDITION _____
STRESSOR INVOLVED INJURY _____
PPE INVOLVED INJURY _____

M/FORM: INJURY LOG ENTRY (#601)

I. PRE-FORM CODE

```
n file, rptfile, filing s: '$d(nocomp) file=ptc(iform,99)
+s: '$d(nocomp) file=ptg(903,iform,99) s New=0, rptfile=1
s LUZ(0)="I"
s U=" " i '$d(DT) s X=$ad($H), $DT=" " d ~DT s DT=Y k X, Y
d sec^tmfm
s U=" " i '$d(DT) s X=$ad($H), $DT=" " d ~DT s DT=Y k X, Y
n file, rptfile, filing s: '$d(nocomp) file=ptc(iform,99)
```

II. WINDOWS

INJURY LOG ENTRY window 0

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
1 INJURY/ILLNESS DATE & TI	DATE	S \$DT="ET" D ~DT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 44

20 CONVERSION CODE:

```
s X=" " s: +$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s: +$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s: $p(cm,".",2) '="" X=X " at " $e(cm,"000",9,10) " " $e(cm,"000",11,12) s X=X " )"
s: X="(1700)" X="" s cm=X
```

28 STORED: \$p(~ICC(DA,0),"",2)

32 POST FILING CODE: d setindx^fmfm

54 INDEX: ~ICC("D")

55 LOOKUP NODE:

```
44~23~$p(~ICC(Lids,0),"",2)
44~23~$p(~ICC(Lids,0),"",2)
43~30~$p(~ICC(Lids,0),"",3)
49~45~$p(~ICC(Lids,100),"",2)
```

M/FORM FID # 61

20 CONVERSION CODE:

28 STORED: \$p(~ICC(DA,0),"",1)

44 POST ID:

```
s: and1 cm=""
s: cm999999999 cm=""
```

2 NAME

FOUNDER 1004' EMPLOY(

M/FORM FID # 43

20 CONVERSION CODE: s cm=\$s(cm="" : "" , '(\$d('EMPLOY(cm,0))#10):cm,1:\$p('EMPLOY(cm,0), "" ,1))

28 STORED: \$p('IOC(DA,0), "" ,3)

30 POSTFIELD CODE:

```

NEW X K X,C,P,FLAG,U,P(0),P(1),P(2),CC
s U="" ,db=$c(26) i $d(new)&(old'=new) s FLAG="" ,P(0)=EMPLOY(new,0)
i $d(FLAG) s:$d(EMPLOY(new,1)) P(1)=EMPLOY(new,1)
i $d(FLAG) S:$d(EMPLOY(new,2)) P(2)=EMPLOY(new,2)
i $d(P(0)) S $P(IOC(DA,0),U,4)=SP(P(0),U,2) ;SSN
I $D(P(0)) S $P(IOC(DA,0),U,5)=SP(P(0),U,3) ;SEX
I $D(P(0)) S $P(IOC(DA,0),U,6)=SP(P(0),U,5) ;BADGE
I $D(P(0)) S $P(IOC(DA,0),U,7)=SP(P(0),U,6) ;SLRTHDAY
I $D(P(0)) S $P(chi(53),db,2)=SP(P(0),U,9) S:chi(53)'="" JO=SP(chi(53),db,2)
I $D(CC) S:CC'="" SP(chi(53),db,1)=SP(DIZ(1001,CC,0),U,2) ;OCCUPATION
I $D(CC) S:CC'="" O=SP(DIZ(1001,CC,0),U,1),chi(52)=C k C ;OCCUP CODE
I $D(P(0)) S $P(IOC(DA,2),U,1)=SP(P(0),U,17) ;AGENCY
I $D(P(0)) S $P(IOC(DA,2),U,2)=SP(P(0),U,12) ;AGENCY UNIT
I $D(P(0)) S $P(IOC(DA,3),U,4)=SP(P(0),U,19) ;BAY PLAN
I $D(P(0)) S X=SP(P(0),U,14) S:X'="" chi(7)=ptc(601,12,7,15,62,X)_db_X K X
I $D(P(1)) S $P(IOC(DA,1),U,1)=SP(P(1),U,2) ;STREET ADDRESS
I $D(P(1)) S $P(IOC(DA,1),U,2)=SP(P(1),U,4) ;CITY
I $D(P(1)) S $P(IOC(DA,1),U,3)=SP(P(1),U,5) ;STREET
I $D(P(1)) S $P(IOC(DA,1),U,6)=SP(P(1),U,6) ;ZIP CODE
I $D(P(2)) S $P(IOC(DA,3),U,5)=SP(P(2),U,1) ;BAY GRADE
I $D(P(2)) S $P(IOC(DA,3),U,3)=SP(P(2),U,5) ;TENURE
I $D(P(2)) S $P(IOC(DA,3),U,6)=SP(P(2),U,2),SP(IOC(DA,3),U,8)=SP(P(2),U,4)
s:'$d(IOC(DA,100)) IOC(DA,100)'="" S:'$d(IOC(DA,200)) IOC(DA,200)'=""
I $D(P(2)) S $P(IOC(DA,3),U,7)=SP(P(2),U,3) ;BAY BASIC
I '$D(DT) S X=$ZD(+SH),DT="" D '$DT S DT=Y
S CL="" S:$d(IOC(DA,0)) CL=SP(IOC(DA,0),U,9)
S:CL="" chi(54)=SZD(+SH) db DT K CL ;INQUIRY LOG ENTRY DATE
S P=0 F %1:1 S P=$O(chi(P)) Q:P="" S:chi(P)'="" chi(0,P)'="" ;filling chi
K FLAG,%P,CC,X,CL,C,U,P(0),P(1),P(2)

```

32 POST FILING CODE: d setindx'fmfm

44 POST ID: i \$d(new) s \$p(chi(54),db,2)=DT

54 INDEX: 'IOC('C')

55 LOOKUP NODE:

```

43~30~$p(IOC(Lids,0), "" ,3)
44~23~$p(IOC(Lids,0), "" ,2)
43~30~$p(IOC(Lids,0), "" ,3)
49~45~$p(IOC(Lids,100), "" ,2)

```

INQUIRY LOG ENTRY Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
1.7 INQUIRY LOG ENTRY DATE	DATE	S '\$DT="E" D '\$DT S X=Y K:YCL X

M/FORM FID # 54

20 CONVERSION CODE:

s X="(" s:+Se(am,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(am,4,5)) " " s:+Se(am,6,7) X=X Se(am,6,7) " " s X=X (1700+Se(am,1,3))
s:Sp(am,".",2)'="" X=X " at " _Se(am "000",9,10) _:" _Se(am "000",11,12) s X=X _)"
s:X="(1700)" X="" s am=X

28 STORED: Sp(^ICC(DA,0),"",9)

12 SHOP

FREE TEXT K:SL(X)>30:(SL(X)<1) X

M/FORM FID # 57

28 STORED: Sp(^ICC(DA,2),"",3)

44 POST ID: s X=am K:SL(X)>30:(SL(X)<1) X s:'\$d(X) am="" s:am("" am=""

22 SHIFT

SET OF CODES 1:DAY
2:SWING
3:GRAVEYARD

M/FORM FID # 7

20 CONVERSION CODE: s am=Sp(Sp("1:DAY;2:SWING;3:GRAVEYARD;",";"_am ":"",2);","",1),am=Sp
(am,"",1)

28 STORED: Sp(^ICC(DA,3),"",9)

15.5 OCCUPATION CODE

FREE TEXT K:SL(X)>6:(SL(X)<5) X

M/FORM FID # 52

28 STORED: Sp(^ICC(DA,3),"",2)

44 POST ID: s X=am K:SL(X)>6:(SL(X)<5) X s:'\$d(X) am="" s:am("" am=""

15 OCCUPATION

FOUNDER 1001' DIZ(1001,

M/FORM FID # 53

20 CONVERSION CODE: s am=\$s(am=""",',(\$d(^DIZ(1001,am,0))#10):am,1:Sp(^DIZ(1001,am,0),"",2))

28 STORED: Sp(^ICC(DA,3),"",1)

32 POST FILING CODE: d setindx^nfim

13 DUTY STATION

FOUNDER 1062' INJ(1062,

M/FORM FID # 58

20 CONVERSION CODE: s am=\$s(am=""",',(\$d(^INJ(1062,am,0))#10):am,1:Sp(^INJ(1062,am,0),"",1))

28 STORED: Sp(^ICC(DA,9),"",1)

23 MONTHS ON JOB

NUMERIC FORMAT K:+X=X!(X>999999999)!(X<0)!(X?.E1)".1N.N) X

M/FORM FID # 8

28 STORED: Sp(^ICC(DA,3),"",11)

402 SUPERVISOR'S NAME FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 62

28 STORED: \$p(^ICC(DA,402),"",1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

4.5 EMPLOYEE COST CENTER FREE TEXT K:\$L(X)>6!(\$L(X)<1) X

M/FORM FID # 56

28 STORED: \$p(^ICC(DA,2),"",4)

44 POST ID: s X=cm K:\$L(X)>6!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

27 SECONDARY SUPERVISOR FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 12

28 STORED: \$p(^ICC(DA,3),"",15)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

107 NONFATAL LOST WORKDAYS SET OF CODES Y:YES

N:NO

M/FORM FID # 9

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";","_cm_":",2)",";","1),cm=\$p(cm,";",1)

28 STORED: \$p(^ICC(DA,100),"",8)

2.6 SUPERVISORY ACC REPORT SET OF CODES Y:YES

N:NO

M/FORM FID # 66

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";","_cm_":",2)",";","1),cm=\$p(cm,";",1)

28 STORED: \$p(^ICC(DA,402),"",7)

112 TYPE OF ACCIDENT POINTIER 1025' INJ(1025,

M/FORM FID # 67

20 CONVERSION CODE: s cm=\$s(cm="":"","(\$d(^INU(1025,cm,0))#10):cm,1:\$p(^INU(1025,cm,0),"",2))

28 STORED: \$p(^ICC(DA,100),"",13)

32 POST FILING CODE: d setindx^fmfm

50 WITNESS NAME

M/FORM FID # 55

28 STORED: nowhere

29 PREFIELD CODE:

;;'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d filecne^ptrfilel k shhfile

i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,l:filing)

+d upontr^fmfm s New=0,cbi(0)=" k file

s lvl=lvl+1

i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

k DA(0) S DA="",D1=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: WITNESS INFORMATION (#602)

M/FORM: WITNESS INFORMATION (#602)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=i
s U="" i '\$d(DT) s X=\$ad(\$H),%DT="" d %DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm

II. WINDOWS

WITNESS INFORMATION WINDOW 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 WITNESS NAME	FREE TEXT	K:\$L(X)>25!(\$L(X)<3) X

M/FORM FID # 6

28 STORED: \$p(^ICC(DA(1),4,DA,0),"",1)
44 POST ID: s X=cm K:\$L(X)>25!(\$L(X)<3) X s:\$d(X) cm="" s:cm["" cm=""

1 STREET ADDRESS	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
------------------	-----------	--------------------------

M/FORM FID # 2

28 STORED: \$p(^ICC(DA(1),4,DA,0),"",2)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

2 CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
--------	-----------	--------------------------

M/FORM FID # 3

28 STORED: \$p(^ICC(DA(1),4,DA,0),"",3)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

3 STATE	POINTER 1003' DIZ(1003,
---------	-------------------------

M/FORM FID # 4

20 CONVERSION CODE: s cm=\$s(cm="":'(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^ICC(DA(1),4,DA,0),"",4)

4 ZIP CODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 5

28 STORED: \$p(^ICC(DA(1),4,DA(0),"",5)

44 POST ID: s X=am K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) am="" s:am("" am=""

6 WITNESS STATEMENT

M/FORM FID # 7

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
i \$d(file),file s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d node0^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D1=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D2 s lvl=lvl-1

16 CALLS M/FORM: WITNESS STATEMENT (#722)

M/FORM: WITNESS STATEMENT (#722)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=~ptc(iform,99)
+s:\$d(nocomp) filen=~ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),%DT="" d ~%DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm

II. WINDOWS

witness statement window 1

DATA ELEMENT

DATA TYPE

FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION

.01 WITNESS STATEMENT

WORD PROCESSING

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(2),4,DA(1),1,DA(0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0`fmfm

120 CAUSE OF INJURY POINTER 1026' INJ(1026,

M/FORM FID # 68

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1026,cm,0))#10):cm,1:\$p(`INJ(1026,cm,0),"",2))
28 STORED: \$p(`ICC(DA,110),"",5)

101 NATURE OF INJURY CODE POINTER 1017' INJ(1017,

M/FORM FID # 49

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1017,cm,0))#10):cm,1:\$p(`INJ(1017,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",2)
32 POST FILING CODE: d setindx`fmfm

100 BODY PART CODE POINTER 1018' INJ(1018,

M/FORM FID # 48

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1018,cm,0))#10):cm,1:\$p(`INJ(1018,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",1)

126 SOURCE OF INJURY POINTER 1027' INJ(1027,

M/FORM FID # 51

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1027,cm,0))#10):cm,1:\$p(`INJ(1027,cm,0),"",2))
28 STORED: \$p(`ICC(DA,120),"",3)
32 POST FILING CODE: d setindx`fmfm

126.5 SOURCE CODE FREE TEXT K:\$L(X)>6!(\$L(X)<1) X

M/FORM FID # 50

28 STORED: \$p(`ICC(DA,120),"",9)
44 POST ID: s X=cm K:\$L(X)>6!(\$L(X)<1) X s:\$d(X) cm="" s:cm"" cm=""

102 INJURY OSHA CODE POINTER 1057' INJ(1057,

M/FORM FID # 16

20 CONVERSION CODE: s cm-\$s(cm="":",('\$d(`INJ(1057,cm,0))#10):cm,1:\$p(`INJ(1057,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",3)

113 BLDG/SHIP FREE TEXT K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 64

28 STORED: \$p(^IOC(DA,100),"",14)
 44 POST ID: s %cm K:SL(X)>15:(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

.4 SPECIFIC LOCATION FREE TEXT K:SL(X)>40:(SL(X)<1) X

M/FORM FID # 65

28 STORED: \$p(^IOC(DA,100),"",15)
 44 POST ID: s %cm K:SL(X)>40:(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

INQUIRY LOG ENTRY WINDOW 2

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
10 RECORDABLE CASE	SET OF CODES	Y:YES N:NO

M/FORM FID # 18

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"_",1),cm=\$p(cm,"_",1)
 28 STORED: \$p(^IOC(DA,100),"",11)

09 RECORDABLE CASE	SET OF CODES	Y:YES N:NO R:REMOVED
--------------------	--------------	----------------------------

M/FORM FID # 22

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;R:REMOVED;"_"_":",2),"_",1),cm=\$p(cm,"_",1)
 28 STORED: \$p(^IOC(DA,100),"",10)

134 RECREATION INQUIRY	SET OF CODES	Y:YES N:NO
------------------------	--------------	---------------

M/FORM FID # 23

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"_",1),cm=\$p(cm,"_",1)
 28 STORED: \$p(^IOC(DA,13),"",3)

111 TERMINATION TRANSFER	SET OF CODES	Y:YES N:NO
--------------------------	--------------	---------------

M/FORM FID # 21

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"_",1),cm=\$p(cm,"_",1)
 28 STORED: \$p(^IOC(DA,100),"",12)

DATE OF DEATH

DATE

S WDT="ET" D WDT S X=Y K:Y<1 X I \$D(X) K:X<\$P(^ICC(DA,0),U,2) X

M/FORM FID # 30

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
e:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10) _:"\$e(cm,"000",11,12) s X=X)"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,100),"",4)

1 AGENT OF ACCIDENT

FOINIER 1068' INJ(1068,

M/FORM FID # 28

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^INJ(1068,cm,0))#10):cm,1:\$p(^INJ(1068,cm,0),"",2))

28 STORED: \$p(^ICC(DA,12),"",4)

3 AGT OF ACC PARTICULAR

FREE TEXT

K:\$L(X)>60!(\$L(X)<1) X

M/FORM FID # 31

28 STORED: \$p(^ICC(DA,12),"",5)

44 POST ID: s X=cm K:\$L(X)>60!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

9 UNSAFE PRACTICE

FREE TEXT

K:\$L(X)>40!(\$L(X)<1) X

M/FORM FID # 32

28 STORED: \$p(^ICC(DA,12),"",6)

44 POST ID: s X=cm K:\$L(X)>40!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

10 UNSAFE CONDITION

FREE TEXT

K:\$L(X)>40!(\$L(X)<1) X

M/FORM FID # 33

28 STORED: \$p(^ICC(DA,12),"",7)

44 POST ID: s X=cm K:\$L(X)>40!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

12 STRESSOR INVOLVED INJURY

FOINIER 1083' STRESS(0,

M/FORM FID # 34

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^STRESS(0,cm,0))#10):cm,1:\$p(^STRESS(0,cm,0),"",1))

28 STORED: \$p(^ICC(DA,13),"",1)

31 PPE INVOLVED INJURY

FREE TEXT

K:\$L(X)>40!(\$L(X)<1) X

M/FORM FID # 35

28 STORED: \$p(^ICC(DA,12),"",8)

44 POST ID: s X=cm K:\$L(X)>40!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

III. POST-PROCESSING CODE

```
i $d(file),file,New=0 s:'$d(lvl) lvl=0
+s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr"from S U=""
```

4.4.2 Enter Speedletter [T3CSPD]

Purpose of transaction: This option uses an M/FORM template for data entry of the Navy Speedletter.

Transaction Description: Menu option initialized when OSHRKS system is installed.

Type of transaction: Input

Global Dictionary Reference: M/FORM ^ptg(903,603

Type of Menu Option: Action

Action/Routine Code: S iform=603,clear=1 K DIC(0),wipe D ^ptrform

Output: Data stored in Injury file #1005 ^ICC(

Retention: Required for historical purposes should be archived.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: M/FORM input templates should be changed by programmer familiar with the VA FileMan -- M/FORM interface problems.

Error Recovery: Error trap by Kernel ^TER routine

Report Format Requirements:

Inputs: See following pages for input templates and documentation.

SPEEDLETTER	
NAME	CASE SERIAL NUMBER
INJURY/ILLNESS DATE & TIME	BLDG/SHIP
SPECIFIC LOCATION	
DUTY STATUS	SHIPS STATUS AT INJURY
MONTHS ON JOB	WORK OPERATION
EVOLUTION OF INJURY	LOST WORK DAYS
NATURE OF INJURY	
EXTENT OF INJURY	
BODY PART	
CAUSE OF INJURY	
TYPE OF ACCIDENT	
WHO CAUSE OF ACCIDENT	
FAILURE TO DO	DAYS HOSPITALIZED
WHY FAILURE	
DATE SPEEDLETTER SENT	
Press <Return> for NARRATIVE	
Press <Return> for CORRECTIVE ACTION	
Press <Return> for SPEEDLETTER REMARKS	

*Word Processing Window will appear

M/FORM: SPEEDLETTER (#603)

I. PRE-FORM CODE

n filen,rptfile,filing s:'\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm

II. WINDOWS

SPEEDLETTER Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	POINTER 1004'	EMPLOY(
--------	---------------	---------

M/FORM FID # 31

20 CONVERSION CODE: s cm=\$e(cm=""": ""',('\$d(EMPLOY(cm,0))#10):cm,1:\$p(EMPLOY(cm,0), """,1))
28 STORED: \$p(ICC(DA,0), """,3)
32 POST FILING CODE: d setindx`fmfm
44 POST ID: i \$D(new) s:'\$d(ICC("C",new,x1)) neg="Person not logged as claim!",cm="",y=24
54 INDEX: ICC("C")
55 LOOKUP NODE:
31^30`\$p(ICC(Lids,0), """,3)
1^23`\$p(ICC(Lids,0), """,2)
36^40`\$p(ICC(Lids,100), """,2)

M/FORM FID # 29

20 CONVERSION CODE:

28 STORED: \$p(ICC(DA,0), """,1)
44 POST ID:
s:cm1 cm=""
s:cm99999999 cm=""

1 INJURY/ILLNESS DATE & TI	DATE	S %DT="ET" D %DT S X=Y K:Y<1 X I \$D(X) R:\$P(X,".",1)>DT X
----------------------------	------	---

M/FORM FID # 1

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEL", ". ",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,0),"",2)
 29 PREFIELD CODE: i '\$d(cbi(31)) s skip="D",endform="Q" ;31*
 32 POST FILING CODE: d setinrk^fmfm
 44 POST ID: i '\$d(new) s:'\$d(^ICC("D",new,x1)) msg="Late not logged as claim!",cm="",y=24
 54 INDEX: ^ICC("D")
 55 LOOKUP NODE:
 1^23^\$p(^ICC(Lids,0),"",2)
 31^30^\$p(^ICC(Lids,0),"",3)
 36^40^\$p(^ICC(Lids,100),"",2)

113 BLDG/SHIP FREE TEXT K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 35

28 STORED: \$p(^ICC(DA,100),"",14)
 44 POST ID: s X=cm K:\$L(X)>15!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

114 SPECIFIC LOCATION FREE TEXT K:\$L(X)>40!(\$L(X)<1) X

M/FORM FID # 34

28 STORED: \$p(^ICC(DA,100),"",15)
 44 POST ID: s X=cm K:\$L(X)>40!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

124 DUTY STATUS SET OF CODES ON:ON DUTY
 OFF:OFF DUTY

M/FORM FID # 9

20 CONVERSION CODE: s cm=\$p(\$p("ON:ON DUTY;OFF:OFF DUTY",";","_cm_":",2)",";"),1),cm=\$p(cm,";",1)
 28 STORED: \$p(^ICC(DA,12),"",1)

118 SHIPS STATUS AT INQUIRY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 7

28 STORED: \$p(^ICC(DA,11),"",3)
 44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

23 MONTHS ON JOB NUMERIC FORMAT K:+X'-X!(X>999999999)!(X<0)!(X?.E1".1N.N) X

M/FORM FID # 10

28 STORED: \$p(^ICC(DA,3),"",11)

117 WORK OPERATION FREE TEXT K:\$L(X)>40!(\$L(X)<1) X

M/FORM FID # 8

28 STORED: \$p(^ICC(DA,11),"",2)
 44 POST ID: s X=cm K:\$L(X)>40!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

115 EVOLUTION OF INJURY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 6

28 STORED: \$p(^ICC(DA,100),"",16)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""

105 LOST WORK DAYS NUMERIC FORMAT K:+X'=X!(X>999)!(X<0)!(X2.EI".IN.N) X

M/FORM FID # 16

28 STORED: \$p(^ICC(DA,100),"",6)

101 NATURE OF INJURY CODE POINTER 1017' INJ(1017,

M/FORM FID # 36

20 CONVERSION CODE: s cm=\$s(cm="":",",'(\$d(^INJ(1017,cm,0))#10):cm,1:\$p(^INJ(1017,cm,0),"",2))

28 STORED: \$p(^ICC(DA,100),"",2)

32 POST FILING CODE: d setindk^fmfm

116 EXTENT OF INJURY SET OF CODES 1:FATALITY

2:PERM TOTAL DISABILITY

3:PERM PARTIAL DISABILITY

4:NO DISABILITY LIKELY

5:MISSING

M/FORM FID # 15

20 CONVERSION CODE: s cm=\$p(\$p("":1:FATALITY;2:PERM TOTAL DISABILITY;3:PERM PARTIAL DISABILITY;
4:NO DISABILITY LIKELY;5:MISSING;"",":_cm_":",2),"",1),cm=\$p(cm,"":",1)

28 STORED: \$p(^ICC(DA,11),"",1)

100 BODY PART CODE POINTER 1018' INJ(1018,

M/FORM FID # 12

20 CONVERSION CODE: s cm=\$s(cm="":",",'(\$d(^INJ(1018,cm,0))#10):cm,1:\$p(^INJ(1018,cm,0),"",2))

28 STORED: \$p(^ICC(DA,100),"",1)

120 CAUSE OF INJURY POINTER 1026' INJ(1026,

M/FORM FID # 13

20 CONVERSION CODE: s cm=\$s(cm="":",",'(\$d(^INJ(1026,cm,0))#10):cm,1:\$p(^INJ(1026,cm,0),"",2))

28 STORED: \$p(^ICC(DA,110),"",5)

112 TYPE OF ACCIDENT POINTER 1025' INJ(1025,

M/FORM FID # 14

20 CONVERSION CODE: s cm=\$s(cm="":",",'(\$d(^INJ(1025,cm,0))#10):cm,1:\$p(^INJ(1025,cm,0),"",2))

28 STORED: \$p(^ICC(DA,100),"",13)

21 WHO CAUSE OF ACCIDENT FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 18

28 STORED: \$p(^ICC(DA,11),"",6)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

22 FAILURE TO DO FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 19

28 STORED: \$p(^ICC(DA,11),"",7)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

19 DAYS HOSPITALIZED NUMERIC FORMAT K:X=X'(X>999)!(X<0)!(X2.E1".1N.N) X

M/FORM FID # 39

28 STORED: \$p(^ICC(DA,110),"",4)

23 WHY FAILURE FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 20

28 STORED: \$p(^ICC(DA,11),"",8)

44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

25 DATE SPEEDLETER SENT DATE S \$DT="E" D \$DT S X=Y K:Y<1 X I \$D(X) K:X<\$P(^ICC(DA,0),U,2) X

M/FORM FID # 38

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,"",2)'="" X=X " at " \$e(cm,"000",9,10)_"_" \$e(cm,"000",11,12) s X=X_"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,120),"",2)

52 NARRATIVE

M/FORM FID # 21

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d fileone^ptrfilel k shhfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,l:filing)
+d upcntr^fcm s New=0,cbi(0)="" k file
s lvl=lvl+i
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(C),DI s lvl=lvl-1

16 CALLS M/FORM: NARRATIVE (#604)

M/FORM: NARRATIVE (#604)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptc(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d "%DT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm
```

II. WINDOWS

NARRATIVE Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NARRATIVE	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('ICC(CA(1),6,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm
```

53 CORRECTIVE ACTION

M/FORM FID # 22

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d filecne`ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-CA,DO-CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-CA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)-CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-CA(i)
s:lvl DA-CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: CORRECTIVE ACTION (#605)

M/FORM: CORRECTIVE ACTION (#605)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocamp) filen='ptc(iform,99)
+s:$d(nocamp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d ~DT s DT=Y k X,Y
d sec~fmfm
d crindx~fmfm
```

II. WINDOWS

CORRECTIVE ACTION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 CORRECTIVE ACTION	WORD PROCESSING	
-----------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),7,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmfm d node0~fmfm
```

54.5 SAFETY REMARKS

M/FORM FID # 30

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone~ptrfile! k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DL=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1
```

16 CALLS M/FORM: SPEEDLETER REMARKS (#723)

M/FORM: SPEEDLETER REMARKS (#723)

I. PRE-FORM CODE

```
n file,rptfile,New s:'$d(nocomp) file=~ptc(iform,99)
+s:$d(nocomp) file=~ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d %DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm
```

II. WINDOWS

SPEEDLETTER REMARKS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 SAFETY REMARKS	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),11,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

III. POST-FORM CODE

```
i $d(file),file,New=0 s:'$d(lvl) lvl=0
+s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fmfm
```

4.4.3 Print/Display Safety Data [T3CSR SAFETY REPORTS]

Purpose of transaction: This menu is used to produce several Safety Office reports, including: OPNAV 5100/10 Log Report, Naval Safety Speedletter, Adjudication Status Report, and Case Summaries.

Transaction Description: This transaction is called from a Menu option [T3C CSR SAFETY REPORTS] in the OSHRKS application package. It is created using VA FileMan print templates and to be used only as an example of reports.

Type of transaction: Print Selection

Global Dictionary Reference:

Type of Menu Option: Menu

Action/Routine Code:

Inputs: Transaction allows person to select from the following.

Retention: This option was created as an example of the type of payment information reports that the site manager can create. Options can be added or deleted at the discretion of the site manager.

Security/Privacy: Calls ^T2USER to verify agency access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^3ER routine

Report Format Requirements:

4.4.3.1 Injury Log Report [T3CSR1 INJURY LOG]

Purpose of transaction: Print/Display Case Summary Report by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR INJURY LOG] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [PRINT INJURY LOG]
Sort template [T3CSORT1 DUTY STA/SHOP/DATE]

Output: PRINT OUT SAMPLE

INJURY LOG REPORT

Case	Date Inj	Employee Name	Dept	BDY	NOI	CD	Death	LTA	LWD	LTD	NCW	T/T
MM/DD/YY								MM/DD/YY				
TOTAL												
COUNT												

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel %ER routine

Report Format Requirements:
OUTPUT FROM WHAT FILE: ICC FILE #1005
PRINT TEMPLATE: [PRINT INJURY LOG]
HEADING: INJURY LOG REPORT
READ ACCESS:
WRITE ACCESS:

PRINT	FORMAT
1. CASE SERIAL NUMBER;"Case"	C1;R6
2. NUMDATE(INJURY/ILLNESS DATE & TIME);"Date Inj"	C8
3. NAME;"Employee Name"	C18;L16
4. SIOF;"Dept"	C35;W4!
5. BODY PART CODE;"BDY"	C40
6. NATURE OF INJURY CODE;"NOI"	C44
7. INJURY OSHA CODE;"CD"	C48;W2
8. NUMDATE(DATE OF DEATH);"Death"	C52
9. LOST TIME ACCIDENT;"LTA"	C62
10. LOST WORK DAYS;"LWD"	C66;R3+
11. LIGHT DUTY DAYS;"LTD"	C70;R3+
12. NONFATAL LOST WORKDAYS;"NCW"	C74
13. TERMINATION TRANSFER;"T/T"	C78

4.4.3.2 Event/Case Summary Report [T3CSR2 CASE SUMMARY]

Purpose of transaction: Print/Display Case Summary Report by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR2 CASE SUMMARY] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [CASE SUMMARY REPORT]
Sort template [T3CSORT2 EMPLOYEE]

Output: PRINT OUT SAMPLE

CASE SUMMARY

Case No: _____ Name/SSN/Bdge: _____/_____
Inj Date/Time: _____ NOI: _____ BP: _____ Shop: _____ Sft: _____
Cost Ctr: _____ Supv: _____ LTA Days: _____ Lt Duty: _____
OWCP File: _____ 3rd Pty Case: _____ DATE to OWCP: _____ ID: _____
Claim No: _____ Form: _____ Date of Claim: MM/DD/YY

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel AFTER routine

Report Format Requirements:
OUTPUT FROM WHAT FILE: ICC FILE #1005
PRINT TEMPLATE: [CASE SUMMARY REPORT]
HEADING: CASE SUMMARY
READ ACCESS:
WRITE ACCESS:

PRINT

FORMAT

1. "Case No:"	S1
2. CASE SERIAL NUMBER;" "	L8
3. "Name/SSN/Bdge:"	C20
4. NAME	X;L25
5. " / "	
6. SSN	X
7. " / "	
8. BADGE NUMBER	X
9. "Inj Date/Time:"	C3
10. NUMDATE(INJURY/ILLNESS DATE & TIME)	X
11. " "	
12. TIME(INJURY/ILLNESS DATE & TIME)	X
13. "NOI:"	C36
14. NATURE OF INJURY CODE	X
15. "BP:"	C45
16. BODY PART CODE	X
17. "Shop:"	C53
13. SHOP	X;L5
19. "Sft:"	C65
20. SHIFT	X
21. "Cost Ctr:"	C3
22. EMPLOYEE COST CENTER	X
23. "Supv:"	C21
24. SUPERVISOR'S NAME	X;L25
25. "LTA Days:"	C53
26. LOST WORK DAYS	X
27. "Lt Duty:"	C69
28. LIGHT DUTY DAYS	X
29. "OWCP File:"	C3
30. OWCP FILE NUMBER	X
31. "3rd Pty Case:"	C30
32. THIRD PARTY CASE	X
33. "DATE to OWCP:"	C49
34. NUMDATE(DATE CLAIM SENT TO OWCP)	X
35. "ID:"	C73
36. CLAIMS CLERK ID	X
37. "Claim No:"	C3
38. SENT TO OWCP	X;L6
39. "Form:"	C20
40. CLAIM FORM LAST RECEIVED	X
41. "Date of Claim:"	C35
42. NUMDATE(DATE OF CLAIM)	X

4.4.3.3 OPNAV 5100/10 Log Report [T3CSR3 OPNAV 5100/10 LOG]

Purpose of transaction: Print/Display OPNAV 5100/10 Report.
Requires a 132 column width.

Transaction Description: This transaction is called from a Menu option [T3CSR3 OPNAV 5100/10 LOG] in the OSHRKS package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Sort template [T3CSORT3 DUTY STA/CASE/DATE]
Print template [OPNAV 5100/10 LOG]

Output: PRINT OUT SAMPLE

OSH LOG

Case No.	Date	Employee Name	Occupation	Department	Description of Injury/Illness		
RECORDABLE CASE:							
		OSHA Death	LTA	LWDays	LtDuty	NFNLW	T/T
	MM/DD/YY						

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel %ER routine

Report Format Requirements:

OUTPUT FROM WHAT FILE: ICC FILE #1005

PRINT TEMPLATE: [OPNAV 5100/10 LOG]

HEADING: OSH LOG

READ ACCESS:

WRITE ACCESS:

PRINT

FORMAT

1. "RECORDABLE CASE:"	S1;C10
2. RECORDABLE CASE	X
3. CASE SERIAL NUMBER;"Case"	R6;C1
4. NUMDATE(INJURY/ILLNESS DATE & TIME);"Date"	C8;W8;Z
5. NAME;"Employee Name"	C17;W16
6. OCCUPATION:OCCUPATION TITLE;"Occupation"	C34;W15;Z
7. SHOP;"Dept"	C50;W4
8. NATURE OF INJURY CODE:NATURE OF INJURY;"Description of"	C55;W
9. BODY PART CODE:BODY PART DESCRIPTION;"Injury/Illness"	C71;W15
10. INJURY OSHA CODE;"OSHA"	C87
11. NUMDATE(DATE OF DEATH);"Death"	C92;Z
12. LOST TIME ACCIDENT;"LTA"	C102
13. LOST WORK DAYS;"LWDays"	C107
14. LIGHT DUTY DAYS;"LtDuty"	C115
15. NONFATAL LOST WORKDAYS;"NFWLW"	C122
16. TERMINATION TRANSFER;"T/T"	C128

4.4.3.4 Personal Case Summary Reports [T3CSR4 PERSONAL CASE SUMMARY]

Purpose of transaction: Print/Display Case Summary Report by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR4 PERSONAL CASE SUMMARY] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [PERSONAL CASE SUMMARY]
Sort template [T3CSORT2 EMPLOYEE]

Output: SAMPLE

PERSONAL INJURY/ILLNESS REPORT

Name:	SSN:	Sex:	DOB: MM/DD/YY
Badge No:	Occupation:		
Case:	Injury Date: MM/DD/YY	Time: 00:00	OSHA Site
Code:	Shop at Inj:	Shift:	Date Entered: MM/DD/YY
	Location:		
Nature of Injury:			
Body Part :			
Accident Type :			
Source :			
Cause of Injury :			
Primary Supervisor :			
Witness Name :			
Date of Death: MM/DD/YY	LTA Days:	Light Duty Days:	
Extent of Injury:		Speedletter Sent: MM/DD/YY	

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel %ER routine

Report Format Requirements:

OUTPUT FROM WHAT FILE: ICC FILE #1005
 PRINT TEMPLATE: [PERSONAL CASE SUMMARY]
 HEADING: PERSONNEL INJURY/ILLNESS REPORT
 READ ACCESS:
 WRITE ACCESS:

	<u>PRINT</u>	<u>FORMAT</u>
1. "Name:"		S1;C1
2. NAMEX;		L25
3. "SSN:"		C35
4. SSN		X
5. "Sex:"		C50
6. SEX		X
7. "DOB:"		C65
8. NUMDATE(BIRTHDATE)		X
9. "Badge No:"		C7
10. BADGE NUMBER		X
11. "Occupation:"		C30
12. OCCUPATION		X
13. " / "		
14. OCCUPATION:OCCUPATION TITLE		X
15. "Case:"		C1
16. CASE SERIAL NUMBER		X;L8
17. "Injury Date:"		C7
18. NUMDATE(INJURY/ILLNESS DATE & TIME)		X
19. "Time:"		C30
20. TIME(INJURY/ILLNESS DATE & TIME)		X
21. "OSHA Site Code:"		C50
22. OSHA SITE CODE		X
23. "Shop at Inj:"		C7
24. SHOP		X;L20
25. "Shift:"		C40
26. SHIFT		X
27. "Date Entered:"		C60
28. NUMDATE(INJURY LOG ENTRY DATE)		X
29. "Location:"		C7
30. SPECIFIC LOCATION		X
31. DUP(" ",50)		X;C15
32. "Nature of Injury:"		C1
33. NATURE OF INJURY CODE		X
34. NATURE OF INJURY CODE:NATURE OF INJURY		X;C30
35. "Body Part:"		C1
36. BODY PART CODE		X
37. BODY PART CODE:BODY PART DESCRIPTION		X;C30
38. "Accident Type:"		C1
39. TYPE OF ACCIDENT		X;L5
40. TYPE OF ACCIDENT:ACCIDENT TYPE		X;C30
41. "Source:"		C1
42. SOURCE OF INJURY		X
43. SOURCE OF INJURY:DESCRIPTION		X;C30
44. "Cause of Injury:"		C1
45. CAUSE OF INJURY		X;L2
46. CAUSE OF INJURY:DESCRIPTION		X;C30

47. DUP(" ",50)	
48. "Primary Supervisor:"	X;C15
49. PRIMARY SUPERVISOR	C1
50. "Witness Name:"	X
51. WITNESS NAME (MULTIPLE)	C1
A. WITNESS NAME	
52. "Date of Death:"	X;W25;C25
53. NUMDATE(DATE OF DEATH)	C1
54. "LTA Days:"	X
55. LOST TIME ACCIDENT	C30
56. "Light Duty Days:"	X
57. LIGHT DUTY DAYS	C50
58. "Extent of Injury:"	X;L3
59. EXTENT OF INJURY	C1
60. "Speedletter Sent:"	X
61. NUMDATE(DATE SPEEDLETTER SENT)	C50
	X

4.4.3.5 Naval Safety Speedletter [T3CSR5 SPEEDLETTER]

Purpose of transaction: Print/Display Naval Safety Speedletter by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR5 SPEEDLETTER] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [SPEEDLETTER/2]
Sort template [T3CSORT2 EMPLOYEE]

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^KER routine

Output: SAMPLE

NAVAL SPEEDLETTER

OPNAV 5102-1 (PID)

Date: MM/DD/YY

To:

SUBJ: NAVAL SPEEDLETTER 5102-1

ALPHA: 1. Activity UIC: _____
2. Type of Mishap: _____
3. Local DTG of Mishap: MM/DD/YY _____
4. _____
5. Location of Mishap: _____
6. Evolution of Mishap: _____
7. Ship's Status at Mishap: _____

BRAVO: N/A

CHARLIE: 1. _____
2. Occupational Grade: _____
3. Duty Status: _____
4. Job at Mishap: _____
5. Months Experience: _____
6. Medical Diagnosis: _____
7. Extent of Injuries/Illness: _____
8. Estimate of Lost Time: _____
A. Total Lost Workdays: _____
B. Days Hospitalized: _____

DELTA: 1. Basic Cause(s) of Mishap: _____
2. Who Caused Mishap: _____
3. Failure to Do: _____
4. Why Failure: _____

ECHO: NARRATIVE
1. Chain of Events: _____
2. Corrective Action: _____
3. Remarks: _____

FROM: Safety Division, Code 106.2

Report Format Requirements:

OUTPUT FROM WHAT FILE: ICC FILE #1005

PRINT TEMPLATE: [SPEEDLETTER/2]

HEADING: @

READ ACCESS:

WRITE ACCESS:

PRINT

FORMAT

1. "NAVAL SPEEDLETTER"	Y1;C32
2. "OPNAV 5102-1 (PID)"	S1;C1
3. "Date:"	C60
4. DATE SPEEDLETTER SENT	X
5. "To: COMMANDER"	S2;C1
6. "NAVAL SAFETY CENTER"	C6
7. "NAVAL AIR STATION"	C6
8. "NORFOLK, VA 23511"	C6
9. "SUBJ: NAVAL SPEEDLETTER 5102-1"	S2;C1
10. "ALPHA: 1. Activity UIC:"	S2;C1
11. ACTIVITY UIC	X
12. "2. Type of Mishap:"	C11
13. TYPE OF ACCIDENT:ACCIDENT TYPE	X
14. "3. Local DTG of Mishap:"	C11
15. INJURY/ILLNESS DATE & TIME	X
16. "4. "	C11
17. DUTY STATION	X
18. " / "	
19. DUTY STATION ADDRESS	X
20. " / "	
21. DUTY STATION CITY	X
22. " / "	
23. DUTY STATION STATE	X
24. " "	
25. DUTY STATION ZIP CODE	X
26. "5. Location of Mishap:"	C11
27. SPECIFIC LOCATION	X
28. "6. Evolution of Mishap:"	C11
29. EVOLUTION OF INJURY	X
30. "7. Ship's Status at Mishap:"	C11
31. SHIPS STATUS AT INJURY	X
32. "BRAVO: N/A"	S2;C1
33. "CHARLIE: 1. "	S2;C1
34. NAME	X
35. " / "	
36. SSN	X
37. " / "	
38. BIRTHDATE	X
39. ' / "	
40. SEX	X
41. "2. Occupational Grade:"	C11
42. OCCUPATION	X
43. " / "	

44. OCCUPATION:OCCUPATION TITLE	X
45. "3. Duty Status:"	C11
46. DUTY STATUS	X
47. "4. Job at Mishap:"	C11
48. WORK OPERATION	X
49. "5. Months Experience:"	C11
50. MONTHS ON JOB	X;L9
51. "6. Medical Diagnosis:"	C11
52. BODY PART CODE:BODY PART DESCRIPTION	X
53. " / "	
54. NATURE OF INJURY CODE:NATURE OF INJURY	X
55. "7. Extent of Injuries/Illness:"	C11
56. EXTENT OF INJURY	X
57. "8. Estimate of Lost Time:"	C11
58. "A. Total Lost Workdays:"	C15
59. LOST WORK DAYS	X;L3
60. "B. Days Hospitalized:"	C15
61. DAYS HOSPITALIZED	X;L3
62. "DELTA: 1. Basic Cause(s) of Mishap:"	S2;C1
63. CAUSE OF INJURY:DESCRIPTION	X
64. "2. Who Caused Mishap:"	C11
65. WHO CAUSE OF ACCIDENT	X
66. "3. Failure to Do:"	C11
67. FAILURE TO DO	X
68. "4. Why Failure:"	C11
69. WHY FAILURE	X
70. "ECHO: Narrative"	S2;C1
71. "1. Chain of Events:"	C11
72. NARRATIVE	X W40
73. "2. Corrective Action:"	C11
74. CORRECTIVE ACTION	X;W38
75. "3. Remarks:"	C11
76. SAFETY REMARKS	X;W32
77. "FROM: Safety Division, Code 106.2"	S2;C1
78. DUTY STATION	X;C11
79. DUTY STATION ADDRESS	X;C11
80. DUTY STATION CITY	X;C11
81. " , "	
82. DUTY STATION STATE	X
83. " "	
84. DUTY STATION ZIP CODE	X

4.4.3.6 Print Detail of Injuries [T3CSR6 ALL INJURIES]

Purpose of transaction: Print/Display all injuries by shop, Injury Date, and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR6 ALL INJURIES] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [ALL INJURIES REPORT]
Sort template [T3CSORT4 DUTY STA/DATE]

Output: PRINT OUT SAMPLE

ALL INJURIES						
Injury Case Date & Time	Employee Name	Job Title	Shop		Specific Location	Bldg/Ship
<hr/>						
MM/DD/YY						
<hr/>						
Cause of Injury		NOI	BP	LWDays	Primary Supervisor	
<hr/>		<hr/>	<hr/>	<hr/>	<hr/>	
COUNT						

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^TER routine

Report Format Requirements:

OUTPUT FROM WHAT FILE: ICC FILE #1005
PRINT TEMPLATE: [ALL INJURIES REPORT]
HEADING: ALL INJURIES
READ ACCESS:
WRITE ACCESS:

PRINT

FC.MAT

1. CASE SERIAL NUMBER;"Case"	R6
2. NUMDATE(INJURY/ILLNESS DATE & TIME);"Inj Date"	C10
3. TIME(INJURY/ILLNESS DATE & TIME);"& Time"	C20
4. NAME;"Employee Name"	C30;L25
5. OCCUPATION:OCCUPATION TITLE;"Job Title"	C57;L20
6. SHOP;"Shop"	C80;L4!
7. SPECIFIC LOCATION;"Specific Location"	C87;L30
8. BLDG/SHIP;"Bldg/Ship"	C120;L12
9. CAUSE OF INJURY:DESCRIPTION;"Cause of Injury"	C20;W35
10. NATURE OF INJURY CODE;"NOI"	C58
11. BODY PART CODE;"BP"	C65
12. LOST WORK DAYS;"LWDays"	C72
13. PRIMARY SUPERVISOR;"Primary Supervisor"	C87

4.4.3.7 Case Summary for Date/Name Range [T3CSR7 CASE RANGE SUMMARY]

Purpose of transaction: Print/Display Case Summary Report by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR7 CASE RANGE SUMMARY] in the OSHRKS package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [CASE RANGE SUMMARY]
Sort template [T3CSORT5 DATE/NAME]

Output: PRINT OUT SAMPLE

CASE SUMMARY BY DATE/NAME

Case No:	Name/SSN/Bdge:	NOI:	BP:	Shop:	Sft:
Inj Date/Time:	Supv:	LTA Davs:	Lt Duty:		
Cost Ctr:	3rd Pty Case:	FEIB to OWCP:	MM/DD/YY	ID:	

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel %ER routine

Report Format Requirements:

OUTPUT FROM WHAT FILE: ICC FILE #1005
PRINT TEMPLATE: [CASE RANGE SUMMARY]
HEADING: CASE SUMMARY BY DATE/NAME
READ ACCESS:
WRITE ACCESS:

<u>PRINT</u>	<u>FORMAT</u>
1. "Case No:"	S1;C1
2. CASE SERIAL NUMBER	X;L8
3. "Name/SSN/Bdge:"	C20
4. NAME	X;L15
5. " / "	
6. SSN	X
7. " / "	
8. BADGE NUMBER	X
9. "Inj Date/Time:"	C3
10. NUMDATE(INJURY/ILLNESS DATE & TIME)	X
11. TIME(INJURY/ILLNESS DATE & TIME)	X
12. "NOI:"	C37
13. NATURE OF INJURY CODE	X
14. "BP:"	C46
15. BODY PART CODE	X
16. "Shop:"	C54
17. SHOP	X;L4
18. "Sft:"	C66
19. SHIFT	X
20. "Cost Ctr:"	C3
21. EMPLOYEE COST CENTER	X
22. "Supv:"	C21
23. PRIMARY SUPERVISOR	X;L26
24. "LTA Days:"	C50
25. LOST WORK DAYS	X;L3
26. "Lt Duty:"	C66
27. LIGHT DUTY DAYS	X;L3
28. "OWCP File:"	C3
29. OWCP FILE NUMBER	X
30. "3rd Party Case:"	C30
31. THIRD PARTY CASE	X
32. "FEHB to OWCP:"	C50
33. NUMDATE(DATE TRANSFER FEHB TO OWCP)	X
34. "ID:"	C73
35. CLAIMS CLERK ID	X

4.4.3.8 Tickler File Print/Display [T3CSR8 TICKLER]

Purpose of transaction: Print/Display Case Summary Report by Injury Date and Employee.

Transaction Description: This transaction is called from a Menu option [T3CSR8 TICKLER] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [TICKLER REPORT2]

Output: PRINT OUT SAMPLE

TICKLER REPORT

MM/DD/YY TIME PAGE #

CSN: _____	SHOP: _____
NAME: _____	SSN: _____
DATE OF INJURY: MM/DD/YY	
CURRENT STATUS: _____	DATE COP BEGAN: MM/DD/YY
TOTAL LWDays: _____	DATE RETURNED TO WORK: _____

TOTAL
COUNT

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^TER routine

Report Format Requirements:

PRINT TEMPLATE: [TICKLER REPORT2]
HEADING: TICKLER REPORT
READ ACCESS:
WRITE ACCESS:

PRINT

1. "CSN:"
2. CASE SERIAL NUMBER
3. "SHOP:"
4. AGENCY UNIT
5. "NAME:"
6. NAME
7. "SSN:"
8. SSN
9. "DATE OF INJURY:"
10. INJURY/ILLNESS DATE & TIME
11. "CURRENT STATUS:"
12. "DATE COP BEGAN:"
13. COP BEGIN DATE
14. "TOTAL LWDays:"
15. LOST WORK DAYS
16. "DATE RETURNED TO WORK:"
17. D/T RETURNED TO WORK

FORMAT

C65
X (SUPPRESS HEADER)
C1
X!
S1;C5
X;L30
C35
X
C35
X
C12
C35
X
C13
X&
C35
X

4.4.3.9 Adjudication Status Report [T3CSR9 ADJUDICATION STATUS]

Purpose of transaction: An output template using the VA FileMan print capabilities to show the adjudication status of individual cases and events.

Transaction Description: This transaction is called from a Menu option [T3CSR9 ADJUDICATION STATUS] in the OSHRKS application package.

Type of transaction: Print Selection

Global Dictionary Reference: ICC(

Type of Menu Option: Routine (Print)

Action/Routine Code: DIPT

Inputs: Print template [ADJUDICATION STATUS]

Output: PRINT OUT SAMPLE

ADJUDICATION STATUS REPORT

MM/DD/YY TIME PAGE #

NAME: _____

SSN: _____

CASE NO: _____

NOI: _____

DOI: _____

ADJ STATUS

DATE ENTERED DATE REMOVED

23	NEED INFORMATION - FROM CLAIMANT & MEDIC	MM/DD/YY	MM/DD/YY
22	NEED INFORMATION - FROM MEDICAL SOURCE	MM/DD/YY	MM/DD/YY
24	NEED INFORMATION - FROM EMPLOYER	MM/DD/YY	MM/DD/YY
A03	CLAIM ACCEPTED BY OWCP - COP AUTHORIZED	MM/DD/YY	(PENDING)

Retention: This option was created as an example of the type of payment information reports that the site manager can create.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Can be changed by site manager.

Error Recovery: Error trap by Kernel ^IER routine

Report Format Requirements:

PRINT TEMPLATE: [ADJUDICATION STATUS]

HEADING: ADJUDICATION STATUS REPORT

READ ACCESS: @

WRITE ACCESS: @

PRINT

FORMAT

1. "NAME:"	S1
2. NAME	X (SUPPRESS HEADER)
3. "SSN:"	C50
4. SSN	X
5. "CASE NO:"	C3
6. CASE SERIAL NUMBER	X;L8
7. "NOI:"	C21
8. NATURE OF INJURY CODE:NATURE OF INJURY	X
9. "DOI:"	C60
10. NUMDATE(INJURY/ILLNESS DATE & ILLNESS)	X
11. "ADJ STATUS"	S1
12. "DATE ENTERED"	C50
13. "DATE REMOVED"	C65
14. DUP("-",80)	X;C1
15. ADJUDICATION STATUS (MULTIPLE)	
A. ADJUDICATION STATUS	X;C1
B. ADJUDICATION STATUS:STATUS DESCRIPTION	X;C5;L40
C. NUMDATE(DATE ENTERED)	X;C50
D. NUMDATE(DATE REMOVED)	X;C65
16. DUP("-",80)	X;C1

4.5 ICC File Maintenance Functions [T2C6 FILE MAINT]. This option allows the module manager to update controlled vocabularies and reference files used exclusively by this module. See Appendix C for list of ICC file attributes. The transactions for this level are as follows:

4.5.1 OWCP Duty Station Agency File — Enter/Edit [T2C6 OWCP AGENCY]

Purpose of transaction: Allows system manager to enter/edit OWCP Duty Station used at the site.

Transaction Description: This transaction is called from a Menu option [T2C6 OWCP AGENCY] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1062

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;9",DIC=1062,DIC(0)="AELMQZ"

Outputs: Data is stored in ^INJ(1062,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^IER routine

4.5.2 Reporting Office for OWCP — Enter/Edit [T2C6 OWCP REPT OFFICE]

Purpose of transaction: Allows system manager to enter/edit OWCP Reporting Office used at the site.

Transaction Description: This transaction is called from a Menu option [T2C6 OWCP REPT OFFICE] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1063

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;8",DIC=1063,DIC(0)="AEMLZ"

Outputs: Data is stored in ^INJ(1063,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel %ER routine

4.5.3 Claim Type File — Enter/Edit [T2C6 CLAIM TYPE EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 CLAIM TYPE EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1029

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs:DR=".01:1",DIC=1029,DIC(0)="-AELMZ"

Outputs: Data is stored in ^INJ(1029,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel %ER routine

4.5.4 Nature of Injury File — Enter/Edit [T2C6 NOI EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 NOI EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1017

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1017,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1017,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^%ER routine

4.5.5 Body Part File — Enter/Edit [T2C6 BODY PART EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 BODY PART EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^F)(1016

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1018,DIC(0)="AEM LZ"

Outputs: Data is stored in ^INJ(1018,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^%ER routine

4.5.6 Accident/Injury Type File — Enter/Edit [T2C6 ACCIDENT FILE EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 ACCIDENT FILE EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1025

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1025,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1025,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^VER routine

4.5.7 Agent of Accident File — Enter/Edit [T2C6 AGENT ACCIDENT EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 AGENT ACCIDENT EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1068

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1068,DIC(0)="AELMC"

Outputs: Data is stored in ^INJ(1068,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^VER routine

4.5.8 Cause of Injury File — Enter/Edit [T2C6 CAUSE INJ EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 CAUSE INJ EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1026

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1026,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1026,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^ER routine

4.5.9 Source of Injury File — Enter/Edit [T2C6 SOURCE INJURY EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 SOURCE INJURY EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1027

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1027,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1027,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^%ER routine

4.5.10 Adjudication Status File -- Enter/Edit [T2C6 ADJUD STATUS EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 ADJUD STATUS EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1061

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1061,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1061,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel ^%ER routine

4.5.11 OSH Injury Codes File -- Enter/Edit [T2C6 OSH INJURY CODES EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 OSH INJURY CODES EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1057

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1057,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1057,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel %ER routine

4.5.12 Pay Status File — Enter/Edit [T2C6 PAY STATUS EDIT]

Purpose of transaction: Allows system manager to enter/edit file.

Transaction Description: This transaction is called from a Menu option [T2C6 PAY STATUS EDIT] in the OSHRKS application package.

Type of transaction: Routine

Global Dictionary Reference: VA FileMan Data Dictionary ^DD(1093

Type of Menu Option: Routine (EDIT)

Action/Routine Code: DIE (See VA FileMan Programmer's Guide)

Inputs: DR=".01;1",DIC=1093,DIC(0)="AELMZ"

Outputs: Data is stored in ^INJ(1093,

Retention: Data can be edited and updated, FileMan file is permanent.

Security/Privacy: Kernel security routines are called to verify read, write, delete, and laygo (add) access.

Flexibility: Table file used in module, can be changed by appropriate authorization.

Error Recovery: Error trap by Kernel %ER routine

APPENDIX A
M/FORM DOCUMENTATION FOR CLAIMS FORMS

APPENDIX A
M/FORM DOCUMENTATION FOR CLAIMS FORMS

FORM LABEL	NAME OF FORM	M/FORM ID #
CA-1	Federal Employee's Notice of Traumatic Injury and Claim for Continuation of Pay/Compensation	613
CA-2	Notice of Occupational Disease and Claim for Compensation	644
CA-2A	Notice of Employee's Recurrence of Disability and Claim for Pay/Compensation	651
CA-3	Report of Termination of Disability and/or Payment	657
CA-5	Claim for Compensation by Widow, and/or Children	672
CA-5B	Claim for Compensation by Parents, Brothers, Sisters, Grandparents, or Grandchildren	688
CA-6	Official Superior's Report of Employee's Death	658
CA-7	Claim for Compensation on Account of Traumatic Injury or Occupational Disease	671
CA-8	Claim for Continuing Compensation on Account of Disability	662
CA-16	Authorization for Examination and/or Treatment	687
CA-17	Duty Status Report	689
CA-20	Attending Physician's Report	708
CA-20A	Attending Physician's Supplemental Report	711

CA-1 EMPLOYEE'S NOTICE OF TRAUMATIC INJURY CLAIM

EMPLOYEE DATA

1. NAME OF EMPLOYEE _____
2. SOCIAL SECURITY NUMBER _____ 3. DATE OF BIRTH _____
4. SEX _____ 5. HOME TELEPHONE _____
6. GRADE AS OF DATE OF INJURY LEVEL _____ STEP _____
7. EMPLOYEE'S HOME MAILING ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
OVERSEAS _____
8. DEPENDENTS _____

DESCRIPTION OF INJURY

9. PLACE WHERE INJURY OCCURRED _____
10. DATE INJURY OCCURRED _____
11. DATE OF THIS NOTICE _____
12. EMPLOYEE'S OCCUPATION _____
13. CAUSE OF INJURY _____
CAUSE OF INJURY EXPLANATION*Press <Return> for CAUSE OF INJURY*
14. NATURE OF INJURY _____ A. OCCUPATION CODE _____
BODY PART _____ B. TYPE CODE _____
C. SOURCE CODE _____
OWCP USE-NOI CODE _____
Press <Return> for NATURE OF INJURY/ILLNESS

EMPLOYEE SIGNATURE

15. SICK AND/OR ANNUAL LEAVE CLAIMED? _____

WITNESS

16. Press <Return> for WITNESS INFORMATION

SUPERVISOR'S REPORT

17. REPORTING OFFICE _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
OWCP AGENCY CODE _____ OSHA SITE CODE _____
18. EMPLOYEE'S DUTY STATION _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
19. REGULAR WORK HOURS _____
20. REGULAR WORK SCHEDULE _____
21. DATE OF INJURY _____
22. DATE NOTICE RECEIVED _____
23. DATE STOPPED WORK _____
24. DATE PAY STOPPED _____
25. DATE 45 DAY PERIOD BEGAN _____
26. DATE RETURNED TO WORK _____
27. EMPLOYEE INJURED IN PERFORMANCE OF DUTY? _____
Press <Return> for EXPLAIN INJURY
28. EMPLOYEE CAUSED INJURY? _____
Press <Return> for WILLFUL MISCONDUCT EXPLANATION
29. WAS INJURY CAUSED BY THIRD PARTY? _____

CA-1 SUPERVISOR'S REPORT CON'T

30. Press <Return> for NAME/ADDRESS THIRD PARTY
31. NAME OF PHYSICIAN 1ST PROVIDING MEDICAL
ADDRESS
CITY STATE ZIP CODE
32. FIRST DATE MEDICAL CARE RECEIVED
33. EMPLOYEE DISABLED FROM WORK?
34. FACTS AGREE WITH EMPLOYEE/WITNESS?
Press <Return> for EXPLAIN NOT AGREE
35. DOES EMPLOYING AGENCY CONTROVERT COP?
Press <Return> for CONTROVERSION EXPLANATION
36. PAY RATE WHEN WORK STOPPED PER
37. SUPERVISOR'S STATEMENT
NAME OF SUPERVISOR DATE
SUPERVISOR'S TITLE OFFICE PHONE
38. FILING INSTRUCTION

WITNESS INFORMATION

WITNESS NAME
STREET ADDRESS
CITY STATE ZIP CODE
Press <Return> for WITNESS STATEMENT

THIRD PARTY INFORMATION

NAME SETTLEMENT AMOUNT
ADDRESS
CITY STATE ZIP CODE

*Word Processing Window will appear.

M/FORM ON-1 (1613)

I. PRE-FORM CODE

```
n file,rptfile,filing s:'$d(nocomp) file='ptc(iform,99);
+s:$d(nocomp) file='ptg(903,iform,99) s New=0,rptfile=1,ixexist=2 k ident
s U="" i '$d(DT) s X=$d($H),%DT="" d '%DT s DT=Y k X,Y
s DLZ(0)="I"
d sec'fmfrm
```

II. WINDOWS

WI EMPLOYEE DATA

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	POINTER 1004'	EMPLOY(
--------	---------------	---------

M/FORM FID # 115

```
17 EXTERNAL DEFAULT: NAME
18 INTERNAL DEFAULT: NAME
20 CONVERSION CODE: s cm=$c(cm:"":",",'($d('EMPLOY(cm,0))#10):cm,1:$p('EMPLOY(cm,0),
"","",1))
28 STORED: $p('ICC(DA,0),"",3)
32 POST FILING CODE: d setindx'fmfrm
44 POST ID: I $D(rsw) s:'$D('ICC("C",new,x1)) mes="Claim is not logged!",y=25,cm=""
54 INDEX: 'ICC("C")
55 LOOKUP NODE:
115^30^$p('ICC(Lids,0),"",3)
16^23^$p('ICC(Lids,0),"",2)
20^20^$p('ICC(Lids,100),"",2)
```

3 SSN	FREE TEXT	K:\$L(X)>9!(\$L(X)<9)!'(X?9N) X
-------	-----------	---------------------------------

M/FORM FID # 2

```
28 STORED: $p('ICC(DA,0),"",4)
44 POST ID: s X=cm K:$L(X)>9!($L(X)<9)!'(X?9N) X s:'$d(;;) cm="" s:cm("" cm=""
```

6 BIRTHDATE	DATE	S %DT="E" D %DT S X=Y K:Y<1 X
-------------	------	-------------------------------

M/FORM FID # 3

```
20 CONVERSION CODE:
s X="(" s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.
NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'=" X=X " at " $e(cm "000",9,10) " " $e(cm "000",11,12)
s X=X ")"
s:X="(1700)" X="" s cm=X
```

```
28 STORED: $p('ICC(DA,0),"",7)
```

4 SEX	SET OF CODES	F:FEMALE M:MALE
-------	--------------	--------------------

U:UNKNOWN

M/FORM FID # 4

20 CONVERSION CODE: s cm=\$p(\$p("F:FEMALE;M:MALE;U:UNKNOWN;";";_cm":",2);";",1)
,cm=\$p(cm,";",1)
28 STORED: \$p(^ICC(DA,0),"",5)

10 HOME PHONE NUMBER FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 5

28 STORED: \$p(^ICC(DA,1),"",5)
44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

18 PAY GRADE FREE TEXT K:SL(X)>3!(SL(X)<2) X

M/FORM FID # 6

28 STORED: \$p(^ICC(DA,3),"",5)
44 POST ID: s X=cm K:SL(X)>3!(SL(X)<2) X s:\$d(X) cm="" s:cm["" cm=""

19 PAY STEP FREE TEXT K:SL(X)>2!(SL(X)<1) X

M/FORM FID # 7

28 STORED: \$p(^ICC(DA,3),"",6)
44 POST ID: s X=cm K:SL(X)>2!(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

7 STREET ADDRESS FREE TEXT K:SL(X)>64!(SL(X)<1) ::

M/FORM FID # 8

28 STORED: \$p(^ICC(DA,1),"",1)
44 POST ID: s X=cm K:SL(X)>64!(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

8 CITY FREE TEXT K:SL(X)>23!(SL(X)<1)!'(X2.AP) X

M/FORM FID # 9

28 STORED: \$p(^ICC(DA,1),"",2)
44 POST ID: s X=cm K:SL(X)>23!(SL(X)<1)!'(X2.AP) X s:\$d(X) cm="" s:cm
["" cm=""

9 SPACE FORMER 1003' DIZ(1003,

M/FORM FID # 10

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0)
,",",1))
28 STORED: \$p(^ICC(DA,1),"",3)

10.5 ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) ::

M/FORM FID # 11

28 STORED: \$p(^ICC(DA,1),"",6)

44 POST ID: s %cm R:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

9.5 OVERSEAS ADDRESS

FREE TEXT

R:SL(X)>64!(SL(X)<1) X

M/FORM FID # 89

28 STORED: Sp("ICC(DA,1),"",4)

44 POST ID: s %cm R:SL(X)>64!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

300 DEPENDENTS

SET OF CODES

S:SPOUSE

C:CHILDREN

O:OTHER

SC:SPOUSE AND CHILDREN

SO:SPOUSE AND OTHER

SD:SPOUSE, CHILDREN, & OTHER

CO:CHILDREN AND OTHER

N:NONE

M/FORM FID # 120

20 CONVERSION CODE: s cm \$p(\$p("S:SPOUSE;C:CHILDREN;O:OTHER;SC:SPOUSE AND CHILDREN;
SO:SPOUSE AND OTHER;SD:SPOUSE, CHILDREN, & OTHER;CO:CHILDREN AND OTHER;

N:NONE;"", " cm ":"", 2).";", 1), cm \$p(cm, ":", 1)

28 STORED: Sp("ICC(DA,300),"",1)

W2 DESCRIPTION OF INJURY

DATA ELEMENT

DATA TYPE

FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION

114 SPECIFIC LOCATION

FREE TEXT

R:SL(X)>40!(SL(X)<1) X

M/FORM FID # 15

28 STORED: Sp("ICC(DA,100),"",15)

44 POST ID: s %cm R:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

1 INJURY/ILLNESS DATE & TI

DATE

S \$DT="ET" D "NOT S X-Y R:Y<1 X I \$D(X) R:\$P(X,".",1)>DT X

M/FORM FID # 16

20 CONVERSION CODE:

s %(" s:+\$e(cm,4,5) X-X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X-X \$e(cm,6,7) " " s X-X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'="" X-X " at " \$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X-X ")"
s:X="(1700)" X="" s cm-X

28 STORED: Sp("ICC(DA,0),"",2)

32 POST FILING CODE: d setirok fmfcm

54 INDEX: "ICC("D")

55 LOOKUP NODE:

1673~Sp("ICC(Lids,0),"",2)

115730~Sp("ICC(Lids,0),"",3)

20720~Sp("ICC(Lids,100),"",2)

202 DATE OF CLAIM

DATE

S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 17

20 CONVERSION CODE:

s X="(" s:+\$(cm,4,5) X=X \$(JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",
+\$(cm,4,5)) " " s:+\$(cm,6,7) X=X \$(cm,6,7) ", " s X=X (1700+\$(cm,1,3))
s:\$(cm,".",2)'=" X=X " at " \$(cm "000",9,10) ":" \$(cm "000",11,12) s X=X "
s:X="(1700)" X="" s cm=X

28 STORED: \$(^ICC(DA,200),"",3)

15 OCCUPATION

FOINIER 1001' DIZ(1001,

M/FORM FID # 18

20 CONVERSION CODE: s cm=\$(cm="":",',(\$d(^DIZ(1001,cm,0))#10):cm,1:\$(^DIZ(1001,cm,0),"",2))

28 STORED: \$(^ICC(DA,3),"",1)

32 POST FILING CODE: d setindx^fmfm

120 CAUSE OF INJURY

FOINIER 1026' INJ(1026,

M/FORM FID # 19

20 CONVERSION CODE: s cm=\$(cm="":",',(\$d(^INJ(1026,cm,0))#10):cm,1:\$(^INJ(1026,cm,0),"",2))

28 STORED: \$(^ICC(DA,110),"",5)

120.1 CAUSE OF INJURY EXPLANATION

M/FORM FID # 114

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s sthfile=1 d filecns^ptrfile1 k sthfile
i \$d(file),file,New=0 s filing=\$(^\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr^fmfm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S D= "",D1=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

16 CALLS M/FORM: CAUSE OF INJURY (#641)

M/FORM: CAUSE OF INJURY (#641)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),NOT="" d NOT s DT=Y k X,Y

d sec`fmfm
d crinck`fmfm

II. WINDOWS

CAUSE OF INJURY WINDOW 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 CAUSE OF INJURY EXPLAINA	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(`ICC(DA(1),111,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s(`\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

101 NATURE OF INJURY CODE POINTER 1017' INJ(1017,

M/FORM FID # 20

20 CONVERSION CODE: s cm-\$s(cm="":",',(\$d(`INJ(1017,cm,0))#10):cm,1:\$p(`INJ(1017,cm,0),"",2);
28 STORED: \$p(`ICC(DA,100),"",2)
32 POST FILING CODE: d setinck`fmfm
54 INDEX: `ICC("E")
55 LOOKUP NODE:
20`20`\$p(`ICC(lids,100),"",2)
115`30`\$p(`ICC(lids,0),"",3)

15.5 OCCUPATION CODE FREE TEXT K:SL(X)>6:(SL(X)<5) X

M/FORM FID # 71

28 STORED: \$p(`ICC(DA,3),"",2)
29 PREFIELD CODE: i \$d(chi(18)) s FL-\$P(chi(18),cb,1) s:FL'="" chi(71)-\$P(`DIZ(1001,FL,0),U,1)
44 POST ID: s X=cm K:SL(X)>6:(SL(X)<5) X s:\$d(X) cm="" s:cm[""" cm=""

100 BODY PART CODE POINTER 1018' INJ(1018,

M/FORM FID # 21

20 CONVERSION CODE: s cm-\$s(cm="":",',(\$d(`INJ(1018,cm,0))#10):cm,1:\$p(`INJ(1018,cm,0),"",2))
28 STORED: \$p(`ICC(DA,100),"",1)

112.5 TYPE CODE FREE TEXT K:SL(X)>5:(SL(X)<1) X

M/FORM FID # 72

28 STORED: \$p("ICC(DA,100),"",17)
29 PREFIELD CODE: NEW X S X=\$p("ICC(DA,100),U,13) S:X="" cbi(72)=\$p("IND(1025,X,0),U,1) K X
44 POST ID: s X=cm K:SL(X)>51(SL(X)<1) X s:\$d(X) cm="" s:cm(""" cm=""

101.1 NATURE OF INJURY EXPLANA

M/FORM FID # 80

28 STORED: nowhere
29 PREFIELD CODE:
s:\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d fileone`ptrfile1 k shhfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,l:l:filing)
+d upcntr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:l s:\$d(DA(i-1)) DA(i)=DA(i-1)
30 POSTFIELD CODE:
i lvl s DA(0)=DA f i=1:l:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: NATURE OF INJURY/ILLNESS (#642)

M/FORM: NATURE OF INJURY/ILLNESS (#642)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=`ptc(iform,99)
+s:\$d(nocomp) filen=`ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),MD="" d `SDT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm

II. WINDOWS

NATURE OF INJURY/ILLNESS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NATURE OF INJURY EXPLANA	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p("ICC(DA(1),101,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl'<filing:lvl+1,l:l:filing)
+d shuffle`fmfm d node0`fmfm

126.5 SOURCE CODE

FREE TEXT

R:SL(X)>6!(SL(X)<1) X

M/FORM FID # 73

28 STORED: Sp("ICC(DA,120),"",9)

29 PREFIELD CODE: NEW X S X=Sp("ICC(DA,120),U,9) S:X'="" cbi(73)=Sp("IND(1027,X,0),U,1) K X

44 POST ID: s X=cm R:SL(X)>6!(SL(X)<1) X s:'\$d(X) cm="" s:cm"" cm=""

101.2 ONCP USE-NOI CODE

FREE TEXT

R:SL(X)>5!(SL(X)<1) X

M/FORM FID # 74

28 STORED: Sp("ICC(DA,100),"",18)

29 PREFIELD CODE: NEW X S X=Sp("ICC(DA,100),U,2) S:X'="" cbi(74)=Sp("ICC(1017,X,0),U,1) K X

44 POST ID: s X=cm R:SL(X)>5!(SL(X)<1) X s:'\$d(X) cm="" s:cm"" cm=""

301 SICK/ANNUAL LEAVE CLAIMS

SET OF CODES

CCP:CONTINUATION OF PAY AND CD
S/A:SICK AND/OR ANNUAL LEAVE

M/FORM FID # 121

20 CONVERSION CODE: s cm=Sp(\$p";CCP:CONTINUATION OF PAY AND COMPENSATION IF DISABILITY
CONTINUES BEYOND 45 DAYS;S/A:SICK AND/OR ANNUAL LEAVE;";" _cm ":";2);";1),cm=Sp(cm,";",1)

28 STORED: Sp("ICC(DA,300),"",2)

WS WITNESS/SUPERVISOR

DATA ELEMENT

DATA TYPE

FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION

50 WITNESS NAME

M/FORM FID # 91

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s sthfile=1 d fileone=ptrfile1 k sthfile
i \$d(file),file,New=0 s filling=\$s('\$d(filling):lvl+1,lvl'<filling:lvl+1,i:filling)
+d upcntr" fmlcm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=CA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: WITNESS INFORMATION (#602)

M/FORM: WITNESS INFORMATION (#602)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1

s U=" i ' \$d(DT) s %=\$d(SH),MDT=" d "MDT s DT=Y k X,Y
d sec' fufm
d crinck' fufm

II. WINDOWS

WITNESS INFORMATION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 WITNESS NAME	FREE TEXT	K:\$L(X)>25!(\$L(X)<3) X
M/FORM FID # 6		
28 STORED: \$p('ICC(DA(1),4,DA,0),"",1)		
44 POST ID: s %cm K:\$L(X)>25!(\$L(X)<3) X s:'\$d(X) cm="" s:cm("" cm=""		
1 STREET ADDRESS	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 2		
28 STORED: 'X'ICC(DA(1),4,DA,0),"",2)		
44 POST ID: s %cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""		
2 CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
M/FORM FID # 3		
28 STORED: \$p('ICC(DA(1),4,DA,0),"",3)		
44 POST ID: s %cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""		
3 STATE	POINTER 1003' DIZ(1003,	
M/FORM FID # 4		
20 CONVERSION CODE: s cm=\$s(cm="" : "" ,('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))		
28 STORED: \$p('ICC(DA(1),4,DA,0),"",3)		
4 ZIP CODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
M/FORM FID # 5		
28 STORED: \$p('ICC(DA(1),4,DA,0),"",5)		
44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""		
6 WITNESS STATEMENT		
M/FORM FID # 7		
28 STORED: no-where		
29 PREFILED CODE:		
s:'\$d(lvl) lvl=0 i \$d(cbl(0))>9 s shhfile=1 d fileone'ptrfile1 k shhfile i \$d(file),file s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,l:filing)		

```
+d node0^fmfm s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DI=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D2 s lvl=lvl-1
```

16 CALLS M/FORM: WITNESS STATEMENT (#722)

M/FORM: WITNESS STATEMENT (#722)

I. PRE-FORM CODE

```
n file,rptfile,New s:$d(ncomp) filen=^ptc(iform,99)
+s:$d(ncomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),NOT="" d ^NOT s DT=Y k X,Y
d sec^fmfm
d crink^fmfm
```

II. WINDOW'S

witness statement window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 WITNESS STATEMENT	WORD PROCESSING	
-----------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(^IOC(DA(2),4,LA(1),1,DA,0), "",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvb>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvb>filing:lvl,1:filing)
+d node0^fmfm
```

14 REPORTING OFFICE

POINTER 1063' INJ(1063,

M/FORM FID # 99

20 CONVERSION CODE: s cm=6s(cm="":",',(\$d(^INU(1063,cm,0))#10):cm,1:\$p(^INU(1063,cm,0),"",1))
28 STORED: \$p(^ICC(DA,10),"",1)

14.01 PO ADDRESS FREE TEXT K:SL(X)>54:(SL(X)<1) X

M/FORM FID # 100

28 STORED: \$p(^ICC(DA,10),"",2)
44 POST ID: s X=cm K:SL(X)>70:(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

14.02 PO CITY FREE TEXT K:SL(X)>20:(SL(X)<1) X

M/FORM FID # 101

28 STORED: \$p(^ICC(DA,10),"",3)
44 POST ID: s X=cm K:SL(X)>20:(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

14.03 PO STATE POINTER 1003' DIZ(1003,

M/FORM FID # 102

20 CONVERSION CODE: s cm=6s(cm="":",',(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^ICC(DA,10),"",4)

14.04 PO ZIP CODE FREE TEXT K:SL(X)>10:(SL(X)<5) X

M/FORM FID # 103

28 STORED: \$p(^ICC(DA,10),"",5)
44 POST ID: s X=cm K:SL(X)>10:(SL(X)<5) X s:\$d(X) cm="" s:cm["" cm=""

13.1 OACP AGENCY CODE FREE TEXT K:SL(X)>6:(SL(X)<6) X

M/FORM FID # 97

28 STORED: \$p(^ICC(DA,9),"",2)
44 POST ID: s X=cm K:SL(X)>6:(SL(X)<6) X s:\$d(X) cm="" s:cm["" cm=""

13.9 OSHA STATE CODE FREE TEXT K:SL(X)>9:(SL(X)<1) X

M/FORM FID # 98

28 STORED: \$p(^ICC(DA,9),"",10)
44 POST ID: s X=cm K:SL(X)>9:(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

13 DUTY STATION POINTER 1062' INU(1062,

M/FORM FID # 92

20 CONVERSION CODE: s cm=6s(cm="":",',(\$d(^INU(1062,cm,0))#10):cm,1:\$p(^INU(1062,cm,0),"",1))
28 STORED: \$p(^ICC(DA,9),"",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 93

28 STORED: Sp(^ICC(DA,9),"",5)

44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.5 DUTY STATION CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 94

28 STORED: Sp(^ICC(DA,9),"",6)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.6 DUTY STATION STATE FOUNDER 1003' DIEZ(1003,

M/FORM FID # 95

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^DIEZ(1003,cm,0))#10):cm,1:\$p(^DIEZ(1003,cm,0),"",1))

28 STORED: Sp(^ICC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 96

28 STORED: Sp(^ICC(DA,9),"",8)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

302 REGULAR WORK HOURS FREE TEXT K:SL(X)>14!(SL(X)<6) X

M/FORM FID # 27

28 STORED: Sp(^ICC(DA,300),"",3)

44 POST ID: s X=cm K:SL(X)>14!(SL(X)<6) X s:'\$d(X) cm="" s:cm["" cm=""

303 REGULAR WORK SCHEDULE FREE TEXT K:SL(X)>15!(SL(X)<3) X

M/FORM FID # 28

28 STORED: Sp(^ICC(DA,300),"",4)

29 PREFIELD CODE:

s:'\$d(cbi(28)) Sp(cbi(28),cb,1)="M-F",cbi(0,28)="",newext="M-F"

i \$d(cbi(28)) s:\$p(cbi(28),cb,1)=" Sp(cbi(28),cb,1)="M-F",cbi(0,28)="

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<3) X s:'\$d(X) cm="" s:cm["" cm=""

M/FORM FID # 69

20 CONVERSION CODE:

28 STORED: nowhere

44 POST ID:

304 DATE RECEIVED NOTICE

DATE

S NOT="ET" D NOT S X-Y K:Y<1 X

M/FORM FID # 29

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm "000",9,10) _:" _Se(cm "000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,300),"",5)

229 D/T STOPPED WORK

DATE

S NOT="ET" D NOT S X-Y K:Y<1 X

M/FORM FID # 83

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm "000",9,10) _:" _Se(cm "000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,221),"",1)

230 D/T PAY STOPPED

DATE

S NOT="ET" D NOT S X-Y K:Y<1 X

M/FORM FID # 81

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm "000",9,10) _:" _Se(cm "000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,221),"",2)

210 COP BEGIN DATE

DATE

S NOT="ET" D NOT S X-Y K:Y<1 X

M/FORM FID # 84

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm "000",9,10) _:" _Se(cm "000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,220),"",1)

29 PREFIELD CODE:

i \$d(cbi(84)) s:\$d(cbi(16))&(cbi(84)="" cbi(84)=cbi(16))
i '\$d(cbi(84))&(\$d(cbi(16))) s cbi(84)=cbi(16)

231 D/T RETURNED TO WORK

DATE

S NOT="ET" D NOT S X-Y K:Y<1 X

M/FORM FID # 85

20 CONVERSION CODE:

```
s %="(" s:+$e(cm,4,5) %X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+$e(cm,4,5)) " " s:+$e(cm,6,7) %X $e(cm,6,7) ", " s %X (1700+$e(cm,1,3))
s:$p(cm,".",2) %X " at " $e(cm "000",9,10) ":" $e(cm "000",11,12) s %X "_"
s:%="(1700)" %X " 3 cmX
```

28 STORED: \$p("IOC(DA,221),"",3)

WM SUPERVISOR PG 2

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
307 INSURED IN LINE OF DUTY	SET OF CODES	Y:YES N:NO

M/FORM FID # 34

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",";","_cm_":",2)",";"),1),cm=\$p(cm,";",1)
28 STORED: \$p("IOC(DA,300),"",8)

307.1 EXPLAIN INQUIRY

M/FORM FID # 35

28 STORED: nowhere

29 PREFIELD CODE:

```
i $d(cbi(34)) s skip="D",cbi(0,35)=""
i $d(cbi(34)) s:cbi(34){"Y"}{"Y"} skip="D",cbi(0,35)=""
s:$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d filecm=ptrfile k shhfile
i $d(file),file,New=0 s filing=$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr fmfma s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=CA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(C)=CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DL s lvl=lvl-1
```

16 CALLS M/FORM: EXPLAIN INQUIRY (#616)

M/FORM: EXPLAIN INQUIRY (#616)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=ptr(iform,99)
+:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i $d(DT) s %$d(SH),%DT="" d %DT s DT=Y k X,Y
d sec fmfma
d crlnk fmfma
```

II. WINDOWS

EXPLAIN INQUIRY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 EXPLAIN INQUIRY	WORD PROCESSING	
---------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),306,DA(0),""),1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s(\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

308 WILLFUL MISCONDUCT	SET OF CODES	Y:YES N:NO
------------------------	--------------	---------------

M/FORM FID # 36

20 CONVERSION CODE: s cm-\$p(\$p("Y:YES;N:NO;",";","_cm_":",2),",",1),cm-\$p(cm,"",1)

28 STORED: \$p(^ICC(DA,300),"",9)

308.1 EXPLAIN WILLFUL MISCONDU

M/FORM FID # 37

26 STORED: nowhere

29 PREFIELD CODE:

i '\$d(cbi(36)) s skip="D",cbi(0,37)=""
i \$d(cbi(36)) s:cbi(36){"n"!cbi(36){"N"} skip="D",cbi(0,37)=""
s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d filecm^ptrfilel k shhfile
i \$d(file),file,New=0 s filing-\$s(\$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)-DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)-DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

16 CALLS M/FORM: WILLFUL MISCONDUCT EXPLANATION (#618)

M/FORM: WILLFUL MISCONDUCT EXPLANATION (#618)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=^ptc(ifcom,99)

```

+s:$d(nocomp) file="ptg(903,iform,99) s New=0,rptfile=1
s U=""" i '$d(DT) s X=$d($H),ADT="" d "AD" s DT=X k X,Y
      d sec`fmfm
      d crink`fmfm

```

II. WINDOWS

WILLFUL MISCONDUCT Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 EXPLAIN WILLFUL MISCONDU	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p("ICU(DA(1),301,DA,0),"",1)

III. POST-FORM CODE

```

i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

```

140 THIRD PARTY CASE	SET OF CODES	Y: YES N: NO
----------------------	--------------	-----------------

M/FORM FID # 38

20 CONVERSION CODE: s cm=\$p(\$p("Y: YES;N: NO;",";","_cm_":",2",";"),1),cm=\$p(cm,"",1)
 28 STORED: \$p("ICC(DA,130),"",9)

141.1 THIRD PARTY

M/FORM FID # 61

28 STORED: nowhere

29 PREFIELD CODE:

```

i '$d(cbi(38)) s skip="D",cbi(0,61)="
i '$d(cbi(38)) s:cbi(38)["n"!((cbi(38))["N") skip="D",cbi(0,61)="
s: '$d(lvl) lvl=0 i '$d(cbi(0))>9 s sthfile=1 d filecre`ptrfilel k sthfile
i '$d(file),file,New'=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=CA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

```

16 CALLS M/FORM: NAME/ADDRESS THIRD PARTY (#626)

M/FORM: NAME/ADDRESS THIRD PARTY (#626)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) tilen~ptc(iform,99)
+s:$d(nocomp) filen~ptg(903,iform,99) s New=0,rptfile=1
s U~"" i '$d(DT) s X~$ad($H),~DT~"" d ~DT s DT=Y k X,Y
d sec~fmfm
d crindx~fmfm
```

II. WINDOWS

NAME/ADDRESS THIRD PARTY Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 6		
28 STORED: \$p(~ICC(DA(1),140,DA,0),"",1)		
44 POST ID: s X~cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""		
5 SETTLEMENT AMOUNT	NUMERIC FORMAT	S:X("\$" X~\$P(X,"\$",2) K:+X'-X(X'7.NL"."2N)!(X>9999999.99)!(X<0) X
M/FORM FID # 7		
28 STORED: \$p(~ICC(DA(1),140,DA,0),"",6)		
1 ADDRESS	FREE TEXT	K:\$L(X)>70!(\$L(X)<1) X
M/FORM FID # 2		
28 STORED: \$p(~ICC(DA(1),140,DA,0),"",2)		
44 POST ID: s X~cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""		
2 CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
M/FORM FID # 3		
28 STORED: \$p(~ICC(DA(1),140,DA,0),"",3)		
44 POST ID: s X~cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""		
3 STATE	FOUNDER 1003' DIZ(1003,	
M/FORM FID # 4		
20 CONVERSION CODE: s cm~\$s(cm~""~""~",('\$d(~DIZ(1003,cm,0))#10):cm,1:\$p(~DIZ(1003,cm,0),"",1))		
28 STORED: \$p(~ICC(DA(1),140,DA,0),"",4)		
4 ZIPCODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
M/FORM FID # 5		

28 STORED: \$p(^ICC(DA(1),140,DA,0),"",5)
 44 POST ID: s X=cm K:SL(X)>101(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
 +d node0^bafm

400 ATTENDING PHYSICIAN FREE TEXT K:SL(X)>301(SL(X)<1) X

M/FORM FID # 105

28 STORED: \$p(^ICC(DA,400),"",1)
 44 POST ID: s X=cm K:SL(X)>301(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.1 DOCS ADDRESS FREE TEXT K:SL(X)>541(SL(X)<1) X

M/FORM FID # 106

28 STORED: \$p(^ICC(DA,400),"",2)
 44 POST ID: s X=cm K:SL(X)>451(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.2 DOCS CITY FREE TEXT K:SL(X)>201(SL(X)<1) X

M/FORM FID # 107

28 STORED: \$p(^ICC(DA,400),"",3)
 44 POST ID: s X=cm K:SL(X)>201(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.3 DOCS SINGLE POINTER 1003' DEZ(1003,

M/FORM FID # 108

20 CONVERSION CODE: s cm=\$e(cm:"":'(\$d('DEZ(1003,cm,0))#10):cm,1:\$p('DEZ(1003,cm,0),"",1))
 28 STORED: \$p(^ICC(DA,400),"",4)

400.4 DOCS ZIPCODE FREE TEXT K:SL(X)>101(SL(X)<5) X

M/FORM FID # 109

28 STORED: \$p(^ICC(DA,400),"",5)
 44 POST ID: s X=cm K:SL(X)>101(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

310 FIRST EDW DRIZ DRIZ S WDT="E" D WDT S X=Y K:Y<1 X

M/FORM FID # 40

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 +\$e(cm,4,5)) " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " s X=X (1700+\$e(cm,1,3))

s:\$p(cm,".",2)'="" X=X_ at "_\$e(cm_"000",9,10)_"_"\$e(cm_"000",11,12) s X=X_"
s:X="(1700)" X="" s cm=X

28 STORED: \$p("ICC(DA,303),"",1)

311 DISABLED FOR WORK SET OF CODES Y:YES
N:NO

M/FORM FID # 41

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_"",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p("ICC(DA,303),"",2)

312 SLPS FACTS AGREE WITH EM SET OF CODES Y:YES
N:NO

M/FORM FID # 42

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";"_cm_"",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p("ICC(DA,303),"",3)

312.1 SUPERVISOR FACTS

M/FORM FID # 51

28 STORED: nowhere

29 PREFIELD CODE:

i '\$d(cbi(42)) s skip="D",cbi(0,51)=""
i '\$d(cbi(42)) s:cbi(42){"Y"!cbi(42){"Y"} skip="D",cbi(0,51)=""
s:'\$d(lvl) lvl=0 i '\$d(cbi(0))>9 s shhfile=l d fileone~ptrfilel k shhfile
i '\$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: EXPLAIN NOT AGREE (#620)

M/FORM: EXPLAIN NOT AGREE (#620)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen="ptc(iform,99)
+s:\$d(nocomp) filen="ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$H),TOT="" d "TOT s DT=X k X,Y
d sec~fmfm
d crindx~fmfm

II. WINDOWS

EXPLAIN NOT AGREE WINDOW 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 SUPERVISOR FACTS	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p('IOC(DA(1),304,DA,0),"",1)		

III. POST-FORM CODE

```
i $d(file),file r filing=$s('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fufm d node0`fufm
```

205 CLAIM CONTOVERSED	SET OF CODES	Y-YES N-D
-----------------------	--------------	--------------

M/FORM FID # 44

```
20 CONVERSION CODE: s cm=$p($p("Y:YES;N:NO;",";","_cm_":",2)",";"),1),cm=$p(cm,";",1)
28 STORED: $p('IOC(DA,200),"",6)
```

205.1 CONTOVERSION EXPLANATO

M/FORM FID # 45

28 STORED: nowhere

29 PREFIELD CODE:

```
i 'Sd(cbi(44)) s skip="D",cbi(0,45)=""
i $d(cbi(44)) s:cbi(44) ["n":(cbi(44) ["N"] skip="D",cbi(0,45)=""
s: 'Sd(lvl) lvl=0 i $d(cbi(0))>9 s shfile=1 d filecne`ptrfile1 k shfile
i $d(file),file,New=0 s filing=$s('Sd(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),M s lvl=lvl-1
```

16 ONLY M/FORM: CONTOVERSION EXPLANATION (#608)

M/FORM: CONTOVERSION EXPLANATION (#608)

I. PRE-FORM CODE

```
n filan,rptfile,New s:'Sd(nocomp) filan=`ptc(iform,99)
```

II. WINDOWS

28 STORED: Sp('IOC(DA(1),25,DA,0),'',1)

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,l:filing)
+d shuffle~fncm d node0~fncm
```

38 STOPPED: Sp("IOC(DA,222),"",1)

20 C7A-MPSTON CODE: S ON SP(SP";H:HOURLY;W:WEEKLY;M:MONTHLY;S:BI-MONTHLY;A:ANNUALLY;
D:DAILY;"", " ON ":",2),";",1),ON SP(ON,"":,1)
28 STORED: SP(TOC(DA,222),"",2)

44 POST ID: s %om K:SL(X)>50:(SL(X)<1) X s:'Sd(X) om"" s:om[""" om""

402 SUPERVISOR'S NAME	TYPE TEXT	K: $GL(X) > 0!$ ($GL(X) < 1$) X

M/FORM FID # 110

28 STORED: Sp(IOC(DA,402),"",1)

44 POST ID: s %cm K:SL(X)>301(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

402.4 DATE SUPERVISOR SIGNED DATE S WDT="E" D WDT S X-Y K:Y<1 X

M/FORM FID # 113

20 CONVERSION CODE:

s %="(" s:+Se(cm,4,5) %K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) %K Se(cm,6,7) " " s %K (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" %K " at " Se(cm "000",9,10) " " Se(cm "000",11,12) s %K ")"
s:%="(1700)" %="" s cm=K

28 STORED: Sp(IOC(DA,402),"",5)

402.1 SUPERVISOR'S TITLE FREE TEXT K:SL(X)>301(SL(X)<1) X

M/FORM FID # 111

28 STORED: Sp(IOC(DA,402),"",2)

44 POST ID: s %cm K:SL(X)>301(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

402.2 SUPERVISOR'S PHONE NUMBER FREE TEXT K:SL(X)>141(SL(X)<7) X

M/FORM FID # 112

28 STORED: Sp(IOC(DA,402),"",3)

44 POST ID: s %cm K:SL(X)>141(SL(X)<7) X s:'\$d(X) cm="" s:cm["" cm=""

314 CLAIM FILING ORIGIN SET OF CODES 1:NO LOST TIME AND NO MEDICAL
2:NO LOST TIME, MEDICAL EXPENSE
3:LOST COVERED BY LEAVE, LWOP,

M/FORM FID # 50

20 CONVERSION CODE: s cm=Sp(Sp("1:NO LOST TIME AND NO MEDICAL EXPENSE;
2:NO LOST TIME, MEDICAL EXPENSE INCURRED;3:LOST COVERED BY LEAVE, LWOP, OR
COP;",";" cm ":",2);",";1), cm=Sp(cm,";",1)
28 STORED: Sp(IOC(DA,305),"",4)

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing=6s('\$d(filing):lvl,lvl>filing:lvl,1:filing!
+d upcntr"from
i file=1!(\$d(ENFLAG)) d ON T3CPAF, T3CPLE

CA-2 NOTICE OF OCCUPATIONAL DISEASE AND CLAIM

EMPLOYEE DATA

1. NAME OF EMPLOYEE _____
2. SOCIAL SECURITY NUMBER _____ 3. DATE OF BIRTH _____
4. SEX _____ 5. HOME TELEPHONE _____
6. GRADE AS OF DATE OF LAST EXPOSURE LEVEL _____ STEP _____
7. EMPLOYEE'S HOME MAILING ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
8. DEPENDENTS _____

CLAIM INFORMATION

9. EMPLOYEE'S OCCUPATION _____ OCCUPATION CODE _____
10. WHERE YOU WORKED WHEN ILLNESS OCCURRED _____
11. DATE YOU FIRST BECAME AWARE OF ILLNESS _____
12. DATE FIRST REALIZED DISEASE/ILLNESS CAUSED _____
13. *Press <Return> for EXPLAIN RELATES TO ILLNESS* OWCP USE-NOI CODE _____
14. *Press <Return> for NATURE OF INJURY/ILLNESS* TYPE CODE _____
15. *Press <Return> for AFTER 30 DAYS* SOURCE CODE _____
16. *Press <Return> for EXPLAIN REASON FOR DELAY*
17. *Press <Return> for EXPLAIN MEDICAL REPORTS DELAY*

SUPERVISOR'S REPORT OF OCCUPATIONAL DELAY

SUPERVISOR'S REPORT

19. REPORTING OFFICE _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
- OWCP AGENCY CODE _____ OSHA SITE CODE _____
20. EMPLOYEE'S DUTY STATION _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
21. REGULAR WORK HOURS _____
22. REGULAR WORK SCHEDULE _____
23. PHYSICIAN FIRST PROVIDING MEDICAL CARE _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
24. FIRST DATE MEDICAL CARE RECEIVED _____
25. WAS EMPLOYEE DISABLED FOR WORK? _____
26. DATE EMPLOYEE 1ST REPORTED TO SUPERVISOR _____
27. DATE AND HOUR EMPLOYEE STOPPED WORK _____
28. DATE AND HOUR EMPLOYEE'S PAY STOPPED _____
29. DATE LAST EXPOSED TO CONDITIONS _____
30. DATE RETURNED TO WORK _____
31. *Press <Return> for DESCRIBE NEW DUTIES*
32. WAS INJURY CAUSED BY THIRD PARTY? _____
33. Press <Return> for NAME/ADDRESS THIRD PARTY
- SIGNATURE OF SUPERVISOR _____
34. SUPERVISOR ACCEPT STATEMENT _____
- NAME OF SUPERVISOR _____
- SUPERVISOR'S TITLE _____
- OFFICE PHONE _____ DATE _____

CA-2 CON'T

THIRD PARTY INFORMATION

NAME _____ SETTLEMENT AMOUNT _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

*Word Processing Window will appear.

M/FORM: ON-2 (1644)

I. PRE-FORM CODE

```

S clear=1
n filen,rptfile,filing s:'$d(nocomp) filen=~ptc(iform,99)
+s:$d(nocomp) filen=~ptg(903,iform,99) s New=0,rptfile=1,idesdist=2 k ident
s U="" i '$d(DT) s X=$ad(SH),WDT="" d ~NOT s DT=Y k X,Y
d sec`fmfm

```

II. WINDOWS

ca2 Window 1

DATA ELEMENT	DATA TYPE	FOINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	FOINTER 1004'	EMPLOY(
M/FORM FID # 67		
17 EXTERNAL DEFAULT: NAME		
18 INTERNAL DEFAULT: NAMEPTR		
20 CONVERSION CODE: s cm=\$s(cm=""', '\$d(EMPLOY(cm,0))#10):cm,1:Sp(EMPLOY(cm,0), "", 1))		
28 STORED: Sp(ICC(FA,0), "", 3)		
32 POST FILING CODE: d setindx`fmfm		
44 POST ID: I SD(new) s: '\$d(ICC("C", new, xl)) msg="CLAIM NOT LOGGED!", y=24, cm=""		
54 INDEX: ICC("C")		
55 LOOKUP NODE:		
67 30 Sp(ICC(Lids,0), "", 3)		
14 23 Sp(ICC(Lids,0), "", 2)		
3 SSN	FREE TEXT	K:SL(X)>9!(SL(X)<9)!'(XQ9N) X
M/FORM FID # 2		
28 STORED: Sp(ICC(DA,0), "", 4)		
44 POST ID: s X=cm K:SL(X)>9!(SL(X)<9)!'(XQ9N) X s: '\$d(X) cm="" s:cm("" cm=""		
6 BIRTHDATE	DATE	S NOT="E" D ~NOT S X=Y K:Y<1 X
M/FORM FID # 3		
20 CONVERSION CODE:		
s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ".", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))		
s:Sp(cm, ".", 2) " " X=X " at " Se(cm "000", 9, 10) " " Se(cm "000", 11, 12) s X=X " " s:X="(1700)" X="" s cm=X		
28 STORED: Sp(ICC(DA,0), "", 7)		
4 SEX	SET OF CODES	F:FEMALE

M:MALE
U:UNKNOWN

M/FORM FID # 4

20 CONVERSION CODE: s cm=Sp(;"F:FEMALE;M:MALE;U:UNKNOWN;";";"_cm_":",2);";",1),cm=Sp(cm,"",1)
28 STORED: Sp(^IOC(DA,0),"",5)

10 HOME PHONE NUMBER FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 5

28 STORED: Sp(^IOC(DA,1),"",5)
44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

18 PAY GRADE FREE TEXT K:SL(X)>3!(SL(X)<2) X

M/FORM FID # 6

28 STORED: Sp(^IOC(DA,3),"",5)
44 POST ID: s X=cm K:SL(X)>3!(SL(X)<2) X s:'\$d(X) cm="" s:cm(""" cm=""

19 PAY STEP FREE TEXT K:SL(X)>2!(SL(X)<1) X

M/FORM FID # 37

28 STORED: Sp(^IOC(DA,3),"",6)
44 POST ID: s X=cm K:SL(X)>2!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

7 STREET ADDRESS FREE TEXT K:SL(X)>64!(SL(X)<1) X

M/FORM FID # 7

28 STORED: Sp(^IOC(DA,1),"",1)
44 POST ID: s X=cm K:SL(X)>64!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

8 CITY FREE TEXT K:SL(X)>23!(SL(X)<1)!'(XZ,AP) X

M/FORM FID # 8

28 STORED: Sp(^IOC(DA,1),"",2)
44 POST ID: s X=cm K:SL(X)>23!(SL(X)<1)!'(XZ,AP) X s:'\$d(X) cm="" s:cm(""" cm=""

9 STREET FREE TEXT K:SL(X)>103! DIZ(1003,

M/FORM FID # 9

20 CONVERSION CODE: s cm=C3(cm=""",',(\$d(^DIZ(1003,cm,0))#10):cm,1:Sp(^DIZ(1003,cm,0),"",1))
28 STORED: Sp(^IOC(DA,1),"",3)

10.5 ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 10

28 STORED: \$p(^ICC(DA,1),"",6)
 44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

300 DEPENDENTS

SET OF CODES S:SPOUSE
 C:CHILDREN
 O:OTHER
 SC:SPOUSE AND CHILDREN
 SO:SPOUSE AND OTHER
 SOD:SPOUSE, CHILDREN, & OTHER
 CO:CHILDREN AND OTHER
 N:NONE

M/FORM FID # 68

20 CONVERSION CODE: s cm=\$p(\$p("S:SPOUSE;C:CHILDREN;O:OTHER;SC:SPOUSE AND CHILDREN;
 SO:SPOUSE AND OTHER;SOD:SPOUSE, CHILDREN, & OTHER;CO:CHILDREN AND OTHER;
 N:NONE;"",") cm ":",2),"",1),cm=\$p(cm,"",1)
 28 STORED: \$p(^ICC(DA,300),"",1)

ca2 Window 2

DATA ELEMENT DATA TYPE POINTER GLOBAL REFERENCE OR INPUT VERIFICATION

15 OCCUPATION POINTER 1001' DIZ(1001,

M/FORM FID # 12

20 CONVERSION CODE: s cm=\$s(cm:"":",',"(\$d(^DIZ(1001,cm,0))#10):cm,1:\$p(^DIZ(1001,cm,0),"",2))
 28 STORED: \$p(^ICC(DA,3),"",1)
 32 POST FILING CODE: d setindx^fmfm

15.5 OCCUPATION CODE FREE TEXT K:SL(X)>6!(SL(X)<5) X

M/FORM FID # 13

28 STORED: \$p(^ICC(DA,3),"",2)
 44 POST ID: s X=cm K:SL(X)>6!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

114 SPECIFIC LOCATION FREE TEXT K:SL(X)>40!(SL(X)<1) X

M/FORM FID # 15

28 STORED: \$p(^ICC(DA,100),"",15)
 44 POST ID: s X=cm K:SL(X)>40!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

1 INQUIRY/TRANS DATE & TI DATE S NOT="ET" D "NOT S X=Y K:Y<1 X I \$d(X) K:\$P(X,"",1)>DT X

M/FORM FID # 14

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","",
 +\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
 s:\$p(cm,"",2)'="" X=X " at " \$e(cm,"000",9,10) "':" \$e(cm,"000",11,12) s X=X "
 s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,0),"",2)
 32 POST FILING CODE: d setindx^fmfma
 54 INDEX: ^ICC("D")
 55 LOOKUP NODE:
 14^23^\$p(^ICC(Lids,0),"",2)
 67^30^\$p(^ICC(Lids,0),"",3)

321 DATE 1ST REALIZED DATE S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 16

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
 s:\$p(cm,"",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X "
 s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,303),"",6)

315 RELATES TO ILLNESS

M/FORM FID # 17

28 STORED: nowhere
 29 PREFIELD CODE:
 s:\$d(lvl) lvl=0 i \$d(chi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
 i \$d(file),file,New=0 s filing=\$e(\$d(filing):lvl+1,lvl)<filing:lvl+1,1:filing)
 +d upcntr^fmfma s New=0,chi(0)="" k file
 s lvl=lvl+1
 i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
 k DA(0) S DA="",DI=""

30 POSTFIELD CODE:
 i lvl s DA(0)=DA f i=1 :lvl s:\$d(DA(i)) DA(i-1)=DA(i)
 s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: EXPLAIN RELATES TO ILLNESS (#629)

M/FORM: EXPLAIN RELATES TO ILLNESS (#629)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=^ptc(ifcom,99)
 +s:\$d(nocomp) filen=^ptg(903,ifcom,99) s New=0,rptfile=1
 s U="" i '\$d(UT) s X=\$ad(\$i),^DT="" d ^DT s DT=Y k X,Y
 d sec^fmfma
 d crindx^fmfma

II. WINDOWS

EXPLAIN TO EMPLOYMENT/WHY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 RELATES TO ILLNESS	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),307,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s(' \$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmfm d node0~fmfm

101.2 CACP USE-NOI CODE FREE TEXT K:\$L(X)>5!(\$L(X)<1) X

M/FORM FID # 19

28 STORED: \$p(^ICC(DA,100),"",18)

44 POST ID: s %cm K:\$L(X)>5!(\$L(X)<1) X s: '\$d(X) cm="" s:cm{"" cm=""

101.1 NATURE OF INJURY EXPLAN

M/FORM FID # 18

28 STORED: nowhere

29 PREFIELD CODE:

s: '\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d fileone~ptrfilel k shhfile
i \$d(file),file,New=0 s filing-\$s(' \$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0-DA f i=lvl:-1:1 s: \$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s: \$d(DA(i)) DA(i-1)=DA(i)
s: lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: NATURE OF INJURY/ILLNESS (#642)

M/FORM: NATURE OF INJURY/ILLNESS (#642)

I. PRE-FORM CODE

n filen,rptfile,New s: '\$d(nocomp) filen=ptc(iform,99)
+s: \$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s %\$ad(\$H),NOT="" d ~NOT s DT=Y k X,Y
d sec~fmfm
d crinck~fmfm

II. WINDOWS

NATURE OF INJURY/ILLNESS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 NATURE OF INJURY EXPLAN	WORD PROCESSING	
-----------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('IOC(DA(1),101,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`mfom d mode0`mfom
```

112.5 TYPE CODE

FREE TEXT

K:\$L(X)>5!(\$L(X)<1) X

M/FORM FID # 20

28 STORED: \$p('IOC(DA,100),"",17)

44 POST ID: s X=cm K:\$L(X)>5!(\$L(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

316 WHY FILED AFTER 30 DAYS

M/FORM FID # 22

28 STORED: nowhere

29 PREFIELD CODE:

```
s: '$d(lvl) lvl=0 i $d(chi(0))>9 s shhfile=l d filecm`ptrfilel k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`mfom s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$L(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="" ,DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: AFTER 30 DAYS (#649)

M/FORM: AFTER 30 DAYS (#649)

I. PRE-FORM CODE

n filen,rptfile,New s: '\$d(nocomp) filen=`ptc(iform,99)

```

+s:$d(nocomp) file="ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),DT="" d "DT s DT=Y k X,Y
      d sec`fmfm
      d crinck`fmfm

```

II. WINDOWS

AFTER 30 DAYS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 WHY FILED AFTER 30 DAYS	WORD PROCESSING	
-----------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(`ICC(DA(1),308,DA,0),"",1)

III. POST-FORM CODE

```

i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

```

126.5 SOURCE CODE

FREE TEXT

K:\$L(X)>6!(\$L(X)<1) X

M/FORM FID # 21

28 STORED: \$p(`ICC(DA,120),"",9)

44 POST ID: s X=on K:\$L(X)>6!(\$L(X)<1) X s: '\$d(X) am="" s:am("" am=""

317 REASON FOR STATEMENT DEL

M/FORM FID # 23

28 STORED: nowhere

29 PREFIELD CODE:

```

s: '$d(lvl) lvl=0 i $d(obi(0))>9 s sthfile=1 d fileone`ptrfile1 k sthfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upentr`fmfm s New=0,obi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-DA,D0-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-DA(i-1)
k DA(0) S DA="" ,DI=""

```

30 POSTFIELD CODE:

```

i lvl s DA(0)-DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

```

16 CALLS M/FORM: EXPLAIN REASON FOR DELAY (#650)

M/FORM: EXPLAIN REASON FOR DELAY (#650)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),DT="" d ~DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm
```

II. WINDOWS

EXPLAIN REASON FOR DELAY Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 REASON FOR STRUMENT DELA	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),309,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-6s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

318 WHY MEDICAL REPORTS DELA

M/FORM FID # 24

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone^ptrfile k shhfile
i $d(file),file,New=0 s filing-6s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: EXPLAIN MEDICAL REPORTS DELAY (#635)

M/FORM: EXPLAIN MEDICAL REPORTS DELAY (#635)

I. PRE-FORM CODE

```

n filen,rptfile,New s:'$d(nocomp) filen~ptc(iform,99)
+s:$d(nocomp) filen~ptg(903,iform,99) s New=0,rptfile=1
s U~"" i '$d(DT) s X=$zd($H),%DT~"" d %DT s DT=Y k X,Y
d sec~fmfm
d crindx~fmfm

```

II. WINDOWS

WHY MEDICAL REPORTS DELAY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 WHY MEDICAL REPORTS DELA	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p(^ICC(DA(1),310,DA,0),"",1)		

III. POST-FORM CODE

```

i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmfm d nodes0~fmfm

```

ca2 Window 3

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
14 REPORTING OFFICE	POINTER 1063'	INJ(1063,
M/FORM FID # 66		
20 CONVERSION CODE: s cm=\$s(cm~""~"";'(\$d(^INJ(1063,ca,0))#10):cm,1:\$p(^INJ(1063,cm,0),"",1))		
28 STORED: \$p(^ICC(DA,10),"",1)		
14.01 RO ADDRESS	FREE TEXT	R:SL(X)>54!(\$L(X)<1) X
M/FORM FID # 52		
28 STORED: \$p(^ICC(DA,10),"",2)		
44 POST ID: s X=cm R:SL(X)>70!(\$L(X)<1) X s:\$d(X) cm~"" s:cm~"" cm~""		
14.02 RO CITY	FREE TEXT	R:SL(X)>20!(\$L(X)<1) X
M/FORM FID # 53		
28 STORED: \$p(^ICC(DA,10),"",3)		
44 POST ID: s X=cm R:SL(X)>20!(\$L(X)<1) X s:\$d(X) cm~"" s:cm~"" cm~""		

14.03 PO STAGE

FOINIER 1003' DIZ(1003,

M/FORM FID # 54

20 CONVERSION CODE: s cm=6s(cm="":",',(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))
28 STORED: \$p('ICC(DA,10),"",4)

14.04 PO ZIP CODE

FREE TEXT

K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 55

28 STORED: \$p('ICC(DA,10),"",5)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

13.1 OACP AGENCY CODE

FREE TEXT

K:SL(X)>6!(SL(X)<6) X

M/FORM FID # 50

28 STORED: \$p('ICC(DA,9),"",2)
44 POST ID: s X=cm K:SL(X)>6!(SL(X)<6) X s:'\$d(X) cm="" s:cm[""" cm=""

13.9 OSHA SLE CODE

FREE TEXT

K:SL(X)>9!(SL(X)<1) X

M/FORM FID # 51

28 STORED: \$p('ICC(DA,9),"",10)
44 POST ID: s X=cm K:SL(X)>9!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

13 DUTY STATION

FOINIER 1062' INU(1062,

M/FORM FID # 45

20 CONVERSION CODE: s cm=6s(cm="":",',(\$d('INU(1062,cm,0))#10):cm,1:\$p('INU(1062,cm,0),"",1))
28 STORED: \$p('ICC(DA,9),"",1)

13.4 DUTY STATION ADDRESS

FREE TEXT

K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 46

28 STORED: \$p('ICC(DA,9),"",5)
44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

13.5 DUTY STATION CITY

FREE TEXT

K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 47

28 STORED: \$p('ICC(DA,9),"",6)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

13.6 DUTY STATION STAGE

FOINIER 1003' DIZ(1003,

M/FORM FID # 48

20 CONVERSION CODE: s cm=6s(cm="":",',(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))

28 STORED: Sp(IOC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:SL(X)>10:(SL(X)<5) X

M/FORM FID # 49

28 STORED: Sp(IOC(DA,9),"",8)

44 POST ID: s X=cm K:SL(X)>10:(SL(X)<5) X s:'Sd(X) cm="" s:cm("" cm=""

302 REGULAR WORK HOURS FREE TEXT K:SL(X)>14:(SL(X)<6) X

M/FORM FID # 28

28 STORED: Sp(IOC(DA,300),"",3)

44 POST ID: s X=cm K:SL(X)>14:(SL(X)<6) X s:'Sd(X) cm="" s:cm("" cm=""

303 REGULAR WORK SCHEDULE FREE TEXT K:SL(X)>15:(SL(X)<3) X

M/FORM FID # 38

28 STORED: Sp(IOC(DA,300),"",4)

44 POST ID: s X=cm K:SL(X)>15:(SL(X)<3) X s:'Sd(X) cm="" s:cm("" cm=""

400 ATTENDING PHYSICIAN FREE TEXT K:SL(X)>30:(SL(X)<1) X

M/FORM FID # 61

28 STORED: Sp(IOC(DA,400),"",1)

44 POST ID: s X=cm K:SL(X)>30:(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm=""

400.1 DOCS ADDRESS FREE TEXT K:SL(X)>54:(SL(X)<1) X

M/FORM FID # 62

28 STORED: Sp(IOC(DA,400),"",2)

44 POST ID: s X=cm K:SL(X)>54:(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm=""

400.2 DOCS CITY FREE TEXT K:SL(X)>20:(SL(X)<1) X

M/FORM FID # 63

28 STORED: Sp(IOC(DA,400),"",3)

44 POST ID: s X=cm K:SL(X)>20:(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm=""

400.3 DOCS STAGE FORM/VER 1003' DIZ(1003,

M/FORM FID # 64

30 CREATION DATE: s cm=6s(cm=""",*(Sd(DIZ(1003,cm,0))#10):cm,1:Sp(DIZ(1003,cm,0),"",1))

28 STORED: Sp(IOC(DA,400),"",4)

400.4 DOCS ZIPCODE FREE TEXT K:SL(X)>10:(SL(X)<5) X

M/FORM FID # 65

28 STORED: Sp('ICC(DA,400),'',5)
44 POST ID: s X=cm K:SL(X)>101(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

310 FIRST DPM DIME DIME S NOT="E" D NOT S X=Y K:Y<1 X

M/FOM FID # 39

20 CONVERSION CODE:
s X="(" s:+Se(cm,4,5) X=K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=K Se(cm,6,7) ", " s X=K (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=K " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=K ")"
s:X="(1700)" X="" s cm=K

28 STORED: Sp('ICC(DA,303),'',1)

311 DISPATCHED FOR WORK SET OF CODES Y:YES
N:NO

M/FOM FID # 31

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO;",";"_cm ":"",2);",1),cm=Sp(cm,"",1)
28 STORED: Sp('ICC(DA,303),'',2)

304 DIME RECEIVED NOTICE DIME S NOT="E" D NOT S X=Y K:Y<1 X

M/FOM FID # 40

20 CONVERSION CODE:
s X="(" s:+Se(cm,4,5) X=K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=K Se(cm,6,7) ", " s X=K (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=K " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=K ")"
s:X="(1700)" X="" s cm=K

28 STORED: Sp('ICC(DA,300),'',5)

229 D/T STOPPED WORK DIME S NOT="E" D NOT S X=Y K:Y<1 X

M/FOM FID # 41

20 CONVERSION CODE:
s X="(" s:+Se(cm,4,5) X=K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=K Se(cm,6,7) ", " s X=K (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=K " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=K ")"
s:X="(1700)" X="" s cm=K

28 STORED: Sp('ICC(DA,221),'',1)

230 D/T BAY SLAPPED DIME S NOT="E" D NOT S X=Y K:Y<1 X

M/FOM FID # 42

20 CONVERSION CODE:
s X="(" s:+Se(cm,4,5) X=K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=K Se(cm,6,7) ", " s X=K (1700+Se(cm,1,3))

s:Sp(cm,".",2)'=" X=X_" at "_Se(cm_"000",9,10)"_:"_Se(cm_"000",11,12) s X=X_"
s:X="(1700)" X="" s cmX

28 STORED: Sp("ICC(DA,221),"",2)

ca2 window 4

DATA ELEMENT	DATA TYPE	FOUNTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

319 DATE LAST EXPOSED TO CON	DATE	S EDT="E" D EDT S X=Y K:Y<1 X
------------------------------	------	-------------------------------

M/FORM FID # 32

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X_" at "_Se(cm_"000",9,10)"_:"_Se(cm_"000",11,12) s X=X_"
s:X="(1700)" X="" s cmX

28 STORED: Sp("ICC(DA,303),"",5)

231 D/T RETURNED TO WORK	DATE	S EDT="E" D EDT S X=Y K:Y<1 X
--------------------------	------	-------------------------------

M/FORM FID # 43

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X_" at "_Se(cm_"000",9,10)"_:"_Se(cm_"000",11,12) s X=X_"
s:X="(1700)" X="" s cmX

28 STORED: Sp("ICC(DA,221),"",3)

320 DESCRIBE NEW WORK ASSIGN

M/FORM FID # 33

28 STORED: nowhere

29 PREFIELD CODE:

s:'Sd(lvl) lvl=0 i Sd(dbi(0))>9 s shifile=l d fileone=ptfilel k shifile
i Sd(file),file,New=0 c filing=Se('Sd(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr'filing s New=0,dbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,00=DA f i=lvl:-1:1 s:Sd(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",dl=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:Sd(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),dl s lvl=lvl-1

15 CALLS M/FORM: DESCRIBE NEW DUTIES (#630)

M/FORM: DESCRIBE NEW DUTIES (#630)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),DT="" d ^DT s DT=Y k X,Y
d sec^fufm
d crindx^fufm
```

II. WINDOWS

DESCRIBE NEW DUTIES Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DESCRIBE NEW WORK ASSING	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p(^IOC(DA(1),311,DA,0),"",1)		

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fufm d node0^fufm
```

140 THIRD PARTY CASE	SET OF CODES	Y:YES N:NO
----------------------	--------------	---------------

M/FORM FID # 34

```
20 CONVERSION CODE: s ca=$p($p("Y:YES;N:NO",";","_ca_":",2),";",1),ca=$p(ca,";",1)
28 STORED: $p(^IOC(DA,130),"",9)
```

141.1 THIRD PARTY

M/FORM FID # 35

28 STORED: nowhere

29 POSTFIELD CODE:

```
i '$d(cbi(34)) s skip="D"
i $d(cbi(34)) s:(cbi(34)["N"])(cbi(34)["N"] skip="D"
s:$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d filecm^ptrfilal k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr^fufm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)-CA,D0-CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-CA(i-1)
```

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: NAME/ADDRESS THIRD PARTY (#626)

M/FORM: NAME/ADDRESS THIRD PARTY (#626)

I. PRE-FORM CITE

n filen,rptfile,New s:\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$H),MDT="" d ~MDT s DT=Y k X,Y
d sec~mfum
d crindx~mfum

II. WINDOWS

NAME/ADDRESS THIRD PARTY Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 NAME	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
----------	-----------	------------------------

M/FORM FID # 6

28 STORED: Sp(~IOC(DA(1),140,DA,0),"",1)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

5 SETTLEMENT AMOUNT	NUMERIC FORMAT S:X(" ") X=Sp(X,"",2) K:+X'-X(X'7.NI".2N)!(X>9999999.99)!(X<0) X
---------------------	---

M/FORM FID # 7

28 STORED: Sp(~IOC(DA(1),140,DA,0),"",6)

1 ADDRESS	FREE TEXT	K:SL(X)>70!(SL(X)<1) X
-----------	-----------	------------------------

M/FORM FID # 2

28 STORED: Sp(~IOC(DA(1),140,DA,0),"",2)
44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

2 CITY	FREE TEXT	K:SL(X)>20!(SL(X)<1) X
--------	-----------	------------------------

M/FORM FID # 3

28 STORED: Sp(~IOC(DA(1),140,DA,0),"",3)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

3 STATE	FOUNDER 1003' DTZ(1003,
---------	-------------------------

M/FORM FID # 4

20 CONVERSION CODE: s cm=ss(cm="" : "" , '(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0), "" ,1))
 28 STORED: \$p('ICC(DA(1),140,DA,0), "" ,4)

4 ZIPCODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 5

28 STORED: \$p('ICC(DA(1),140,DA,0), "" ,5)
 44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s: '\$d(X) cm="" s:cm("" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=ss(' \$d(filing):lvl,lvb>filing:lvl,1:filing)
 +d node0" file

720 OFFICIAL REMARKS FREE TEXT K:\$L(X)>50!(\$L(X)<1) X

M/FORM FID # 56

28 STORED: \$p('ICC(DA,402), "" ,6)
 44 POST ID: s %cm K:\$L(X)>50!(\$L(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

402 SUPERVISOR'S NAME FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 57

28 STORED: \$p('ICC(DA,402), "" ,1)
 44 POST ID: s %cm K:\$L(X)>30!(\$L(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

402.1 SUPERVISOR'S TITLE FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 58

28 STORED: \$p('ICC(DA,402), "" ,2)
 44 POST ID: s %cm K:\$L(X)>30!(\$L(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

402.2 SUPERVISOR'S PHONE NUMBER FREE TEXT K:\$L(X)>14!(\$L(X)<7) X

M/FORM FID # 59

28 STORED: \$p('ICC(DA,402), "" ,3)
 44 POST ID: s %cm K:\$L(X)>14!(\$L(X)<7) X s: '\$d(X) cm="" s:cm("" cm=""

402.4 DATE SUPERVISOR SIGNED DATE S "M/Y" D "M/Y" S %Y K:Y<1 X

M/FORM FID # 60

20 CONVERSION CODE:
 s %(" s:+\$e(cm,4,5) %X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", " ,
 +\$e(cm,4,5)) " s:+\$e(cm,6,7) %X \$e(cm,6,7) " , " s %X (1700+\$e(cm,1,3))

s:\$p(cm,".",2)'="" X=X_ at "_\$e(cm "000",9,10) _:"_\$e(cm "000",11,12) s X=X_"
s:X="(1700)" X="" s cmX

28 STORED: \$p(`ICC(DA,402),"",5)

III. POST-FORM CODE

```
i $d(file).file,New=C s:'$d(lvl) lvl=0  
+s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)  
+d upcntr`fmfm  
i file=1!($d(ERFLAG)) d CAL`T3CPNF,T3CPFILE
```

CA-2A NOTICE OF RECURRENCE OF DISABILITY

PART A--EMPLOYER

1. NAME OF INJURED EMPLOYEE _____
2. SOCIAL SECURITY NUMBER _____
3. OWCP FILE NUMBER _____
4. HOME MAILING ADDRESS (INCLUDE ZIP CODE) _____
 CITY _____ STATE _____ ZIP CODE _____
5. HOME TELEPHONE _____
6. NAME OF EMPLOYING ESTABLISHMENT AT TIME OF ORIGINAL INJURY _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
7. NAME OF EMPLOYING ESTABLISHMENT AT TIME OF RECURRENCE, IF OTHER THAN #6 _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
8. DATE AND HOUR OF ORIGINAL INJURY (MO., DAY, YEAR) _____
9. DATE AND HOUR OF RECURRENCE _____
10. DATE/HOUR STOPPED WORK (AFTER RECURRENCE) _____
11. DATE/HOUR PAY STOPPED (AFTER RECURRENCE) _____
12. PAY RATE IN EFFECT ON:

	A. BASE PAY	B. SUBSISTENCE	C. QUARTERS	D. OTHER PAY
A. DATE	\$ _____	\$ _____	\$ _____	\$ _____
RECURRENCE PER	_____	PER _____	PER _____	PER _____
B. DATE WK	\$ _____	\$ _____	\$ _____	\$ _____
STP AFT PER	_____	PER _____	PER _____	PER _____
13. SHOW WORK WEEK AT TIME PAY STOPPED _____
14. DATE/HOUR RETURNED TO WORK (AFTER RECUR) _____
15. WAS MEDICAL TREATMENT AUTHORIZED? _____
16. DATE FIRST RECEIVED MEDICAL TREATMENT _____
17. ATTENDING PHYSICIAN _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
18. DISABLED FOR WORK? _____
 Press <Return> for DESCRIBE NEW DUTIES
19. *Press <Return> for SUPERVISOR'S STATEMENT*
20. SIGNATURE OF OFFICIAL SUPERIOR _____
21. TITLE _____
22. WORK PHONE NUMBER _____
23. DATE _____

PART B--CONTINUATION OF PAY

24. REGULAR PAY CONTINUED DURING THIS PERIOD _____
25. SHOW GROSS OF REGULAR PAY _____
26. IF PAY CHANGED DURING COP, DATE CHANGED _____
27. IF PAY RATE CHANGED, GIVE NEW RATE

A. BASE PAY _____	B. SUBSISTENCE _____	C. QUARTERS _____
D. OTHER (SPECIFY) _____	SPECIFY OTHER _____	

CA-2A CON'T

PART C--EMPLOYEE

28. Press <Return> for WORK INFORMATION

29. WHILE DISABLED FOR WORK:

A. SICK/ANNUAL LEAVE, FROM: THROUGH:
B. CONTINUATION OF PAY, FROM: THROUGH:
C. CONTINUING COMPENSATION, FROM: THROUGH:

31. DATE (MONTH, DAY, YEAR)

WORK INFORMATION

DATES & HOURS WORKED

PAY RATE

TOTAL AMOUNT EARNED

TYPE WORK PERFORMED

EMPLOYER NAME

EMPLOYER STREET ADDRESS

EMPLOYER CITY/STATE/ZIP CODE

*Word Processing Window will appear.

M/FORM: CA-2A (#651)

I. PRE-FORM CODE

```

n filen,rptfile,filing s:'$d(nocomp) filen~ptc(iform,99)
+s:$d(nocomp) filen~ptg(903,iform,99) s New=0,rptfile=1,ident=2 k ident
s U="" i '$d(DT) s X=$d($H),DT="" d ~DT s DT=Y k X,Y
d sec~fmfm
n filen,rptfile,filing s:'$d(nocomp) filen~ptc(iform,99)
I $D(DA) S PAY=$S($D('ICC(DA,231)):^(231),$D('ICC(DA,229)):^(229),1: "")
I $D(DA) S ('ICC(DA,213),'ICC(DA,214))-PAY,'ICC(DA,210)="" K PAY

```

II. WINDOWS

WI EMPLOYER

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	POINTER 1004' EMPLOY(
M/FORM FID # 57		
17 EXTERNAL DEFAULT: NAME		
18 INTERNAL DEFAULT: NAMEPTR		
20 CONVERSION CODE: s cm=\$s(cm~""~""~'(\$d('EMPLOY(cm,0))#10):cm,1:\$p('EMPLOY(cm,0),"",1))		
28 STORED: \$p('ICC(DA,0),"",3)		
32 POST FILING CODE: d setindx~fmfm		
44 POST ID: I \$D(new) s:'\$d('ICC("C",new,x1)) cm~""~msg="CLAIM NOT LOGGED!",y=24		
54 INDEX: 'ICC("C")		
55 LOOKUP NODE:		
57~30~\$p('ICC(1ids,0),"",3)		
12~23~\$p('ICC(1ids,0),"",2)		
3 SSN	FREE TEXT	K:\$L(X)>9!(\$L(X)<9)!'(XQ9N) X
M/FORM FID # 2		
28 STORED: \$p('ICC(DA,0),"",4)		
44 POST ID: s X~cm K:\$L(X)>9!(\$L(X)<9)!'(XQ9N) X s:'\$d(X) cm~"" s:cm[""" cm~""		
137 OACF FILE NUMBER	FREE TEXT	K:\$L(X)>13!(\$L(X)<1) X
M/FORM FID # 3		
28 STORED: \$p('ICC(DA,130),"",6)		
44 POST ID: s X~cm K:\$L(X)>13!(\$L(X)<1) X s:'\$d(X) cm~"" s:cm[""" cm~""		
7 STREET ADDRESS	FREE TEXT	K:\$L(X)>64!(\$L(X)<1) X
M/FORM FID # 4		

28 STORED: \$p(^IOC(DA,1),"",1)
44 POST ID: s %cm K:SL(X)>64!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

8 CITY FREE TEXT K:SL(X)>23!(SL(X)<1)!'(XZ.AP) X

M/FORM FID # 5

28 STORED: \$p(^IOC(DA,1),"",2)
44 POST ID: s %cm K:SL(X)>23!(SL(X)<1)!'(XZ.AP) X s:'\$d(X) cm="" s:cm["" cm=""

9 STATE FOUNDER 1003' DIZ(1003,

M/FORM FID # 6

20 CONVERSION CODE: s cm-\$s(cm=""": "", '(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^IOC(DA,1),"",3)

10.5 ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 7

28 STORED: \$p(^IOC(DA,1),"",6)
44 POST ID: s %cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

9.5 OVERSEAS ADDRESS FREE TEXT K:SL(X)>64!(SL(X)<1) X

M/FORM FID # 8

28 STORED: \$p(^IOC(DA,1),"",4)
44 POST ID: s %cm K:SL(X)>64!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

10 HOME PHONE NUMBER FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 9

28 STORED: \$p(^IOC(DA,1),"",5)
44 POST ID: s %cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13 DUTY STATION FOUNDER 1062' INJ(1062,

M/FORM FID # 32

20 CONVERSION CODE: s cm-\$s(cm=""": "", '(\$d(^INJ(1062,cm,0))#10):cm,1:\$p(^INJ(1062,cm,0),"",1))
28 STORED: \$p(^IOC(DA,9),"",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 33

28 STORED: \$p(^IOC(DA,9),"",5)
44 POST ID: s %cm K:SL(X)>70!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.5 DUTY STATION CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 34

28 STORED: \$p('TOC(DA,9),"",6)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

13.6 DUTY STATION STATE

FOUNDER 1003' DIZ(1003,

M/FORM FID # 35

20 CONVERSION CODE: s cm=\$s(cm=""":""',('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))

28 STORED: \$p('TOC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE

FREE TEXT

K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 36

28 STORED: \$p('TOC(DA,9),"",8)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

2201 DUTY STATION AT RETURN

FOUNDER 1062' INJ(1062,

M/FORM FID # 52

20 CONVERSION CODE: s cm=\$s(cm=""":""',('\$d('INJ(1062,cm,0))#10):cm,1:\$p('INJ(1062,cm,0),"",1))

28 STORED: \$p('TOC(DA,210),"",4)

2201.1 REFERENCE ADDRESS

FREE TEXT

K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 53

28 STORED: \$p('TOC(DA,210),"",5)

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

2201.2 REFERENCE CITY

FREE TEXT

K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 54

28 STORED: \$p('TOC(DA,210),"",6)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

2201.3 REFERENCE STATE

FOUNDER 1003' DIZ(1003,

M/FORM FID # 55

20 CONVERSION CODE: s cm=\$s(cm=""":""',('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))

28 STORED: \$p('TOC(DA,210),"",7)

2201.4 REFERENCE ZIP CODE

FREE TEXT

K:SL(X)>10!(SL(X)<5)!('XGEN(XGEN""-""N)) X

M/FORM FID # 56

28 STORED: \$p('TOC(DA,210),"",8)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5)!('XGEN(XGEN""-""N)) X s:'\$d(X) cm="" s:cm[""" cm=""

1 INJURY/ILLNESS DATE & TIME DATE

S %DT="ET" D %DT S %Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 12

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p("ICC(DA,0),"",2)

32 POST FILING CODE: d setindx~fmfm

54 INDEX: "ICC("D")

55 LOOKUP CODE:

12~23\$P("ICC(Lids,0),"",2)

57~30\$P("ICC(Lids,0),"",3)

2202 DATE/TIME OF RECURRENCE

DATE

S %DT="ET" D %DT S %Y K:Y<1 X

M/FORM FID # 31

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p("ICC(DA,210),"",1)

2203 REC D/T STOPPED WORK

DATE

S %DT="ET" D %DT S %Y K:Y<1 X

M/FORM FID # 67

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p("ICC(DA,210),"",9)

2204 RECURRENT DT PAY STOPPE

DATE

S %DT="ET" D %DT S %Y K:Y<1 X

M/FORM FID # 77

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p("ICC(DA,210),"",10)

W2 CONTINUATION/EMPLOYEE

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2202.1 REFERENCE BASE PAY	NUMERIC FORMAT S:X(5) X=SP(X,"S",2) K:+X'-X&(X'7.NI"."2N)!(X>999999.99)!(X<0) X	
M/FORM FID # 59		
28 STORED: Sp(IOC(DA,214),"",1)		
2202.3 REFERENCE SUBSISTENCE	NUMERIC FORMAT S:X(5) X=SP(X,"S",2) K:+X'-X&(X'7.NI"."2N)!(X>999999.99)!(X<0) X	
M/FORM FID # 60		
28 STORED: Sp(IOC(DA,214),"",3)		
2202.4 REFERENCE QUARTERS	NUMERIC FORMAT S:X(5) X=SP(X,"S",2) K:+X'-X&(X'7.NI"."2N)!(X>999999.99)!(X<0) X	
M/FORM FID # 63		
28 STORED: Sp(IOC(DA,214),"",5)		
2202.5 REFERENCE OTHER RATE	NUMERIC FORMAT S:X(5) X=SP(X,"S",2) K:+X'-X&(X'7.NI"."2N)!(X>999999.99)!(X<0) X	
M/FORM FID # 64		
28 STORED: Sp(IOC(DA,214),"",7)		
2202.2 REFERENCE BASE BASIS	SET OF CODES	H:HOURLY W:WEEKLY M:MONTHLY B:BI-MONTHLY A:ANNUALLY D:DAILY
M/FORM FID # 58		
20 CONVERSION CODE: s cm Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;D:DAILY;";" cm ":"2);";",1),cm Sp(cm,";",1)		
28 STORED: Sp(IOC(DA,214),"",2)		
2202.3 REFERENCE SUB BASIS	SET OF CODES	H:HOURLY W:WEEKLY M:MONTHLY B:BI-MONTHLY A:ANNUALLY D:DAILY
M/FORM FID # 61		
20 CONVERSION CODE: s cm Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;D:DAILY;";" cm ":"2);";",1),cm Sp(cm,";",1)		
28 STORED: Sp(IOC(DA,214),"",4)		

M/FORM FID # 62

2202.55REC OTHER BASIS

M. FORM FID # 65

2203. LREC WORK STOPPED BASE RA ALPHERIC FORMAT S:X1"\$" X=SP(X,"\$".2) K:+X'-X/(X? .N1". "2N)!(X>9999999.99)!(X<0) X

M. FORM FID # 69

2203. REC WORK STOPPED SUBSIST NUMERIC FORMAT S:X1 "S" X:SP(X,"S",2) K:X'-X(X'?.N1". "2V)!(X>9999999.99)!(X<0) X

M. FORM FID # 70

2203. SPEC WORK STOPPED QUARTER NUMERIC FORMAT S:X' "S" X=SP(X,"S",2) K:+X'-XK(X'?.M1",".2N)!!(X>999999.99)!!(X<0) X

M FORM FID # 72

2203. SPEC WORK STOPPED OTHER N NUMERIC FORMAT S:X("S" X=SP(X,"S",2) K:+X'-X(X'?.N1",".2N)/((X0999999.99)/(X<0) X

FORM FID # 74

2303. REC WORK STOPPED BASE BA

A-52

A:ANNUALLY
D:DAILY

M/FORM FID # 68

20 CONVERSION CODE: s cm Sp(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm Sp(cm,":",1)
28 STORED: Sp(TOC(DA,213),""2)

2203.3SPEC WORK STOPPED SUB BAS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 71

20 CONVERSION CODE: s cm Sp(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm Sp(cm,":",1)
28 STORED: Sp(TOC(DA,213),""4)

2203.4SPEC WORK STOPPED QUART BAS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 73

20 CONVERSION CODE: s cm Sp(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm Sp(cm,":",1)
28 STORED: Sp(TOC(DA,213),""6)

2203.5SPEC WORK STOPPED OTHER B SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 75

20 CONVERSION CODE: s cm Sp(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm Sp(cm,":",1)
28 STORED: Sp(TOC(DA,213),""8)

303 REGULAR WORK SCHEDULE FREE TEXT K:SL(X)>151(SL(X)<3) X

M/FORM FID # 17

28 STORED: Sp(TOC(DA,300),""4)
44 POST ID: s X cm K:SL(X)>101(SL(X)<3) X s:'\$d(X) cm"" s:cm[""" cm""

2205 REC N/T RETURNED TO WORK DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 79

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)="-" X=X " at " Se(cm "000",9,10) "-" Se(cm "000",11,12) s X=X "
s:X="(1700)" X="" s cmX

28 STORED: Sp("ICC(DA,210),"",11)

2206 DID OACP AUTHORIZE TREAT SET OF CODES Y:YES
N:NO

M/FORM FID # 19

20 CONVERSION CODE: s cm=Sp("Y:YES;N:NO;n:NO;y:YES;"",":cm ":"",2),"",1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,210),"",2)

2207 DATE 1ST TREATMENT AFTER DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 20

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)="-" X=X " at " Se(cm "000",9,10) "-" Se(cm "000",11,12) s X=X "
s:X="(1700)" X="" s cmX

28 STORED: Sp("ICC(DA,210),"",3)

400 ATTENDING PHYSICIAN FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 42

28 STORED: Sp("ICC(DA,400),"",1)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""" cm=""

400.1 DOCS ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 43

28 STORED: Sp("ICC(DA,400),"",2)
44 POST ID: s X=cm K:SL(X)>45!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""" cm=""

400.2 DOCS CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 44

28 STORED: Sp("ICC(DA,400),"",3)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'Sd(X) cm="" s:cm[""" cm=""

400.3 DOCS STATE POINTER 1003' DIZ(1003,

M/FORM FID # 45

20 CONVERSION CODE: s cm=\$s(cm=""",'(\$d(DIZ(1003,cm,0))#10):cm,1:\$p(DIZ(1003,cm,0),"",1))
28 STORED: \$p(TOC(DA,400),"",4)

400.4 DOCS ZIPCODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 46

28 STORED: \$p(TOC(DA,400),"",5)
44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s: '\$d(X) cm="" s:cm("" cm=""

W3 DUTIES

DATA ELEMENT	DATA TYPE	FORMER GLOBAL REFERENCE OR INPUT VERIFICATION
311 DISABLED FOR WORK	SET OF CODES	Y:YES N:NO

M/FORM FID # 22

20 CONVERSION CODE: s cm=\$p(\$p";Y:YES;N:NO;";" _cm ":";2);";1),cm=\$p(cm,"";1)
28 STORED: \$p(TOC(DA,303),"",2)

320 DESCRIBE NEW WORK ASSIGN

M/FORM FID # 23

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(cbl(22)) skip="D",cbl(0,23)=""
1 \$d(cbl(22)) s:cbl(22){ "N" i(cbl(22)){ "N" skip="D",cbl(0,23)=""
s:'\$d(lvl) lvl=0 1 \$d(cbl(0))>9 s sthfile=l d filecm="prfilal k sthfile
1 \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr"fmcm s New=0,cbl(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=CA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

44 POST ID:

1 '\$d(cbl(22)) s skip="D",cbl(0,23)=""
1 \$d(cbl(22)) s:cbl(22){ "Y" i(cbl(22)){ "Y" skip="D",cbl(0,23)=""

16 CALLS M/FORM: DESCRIBE NEW DUTIES (#630)

M/FORM: DESCRIBE NEW DUTIES (#630)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen="ptc(ifom,99)

```

+s:$d(nocomp) file="ptg(903,iform,99) s New=0,r,file=1
s U="" i '$d(DT) s %$d($H),NOT="" d "NOT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm

```

II. WINDOWS

DESCRIBE NEW DUTIES Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DESCRIBE NEW WORK ASSING	WORD PROCESSING	
------------------------------	-----------------	--

M/FOM FID # 1

28 STORED: \$p('IUC(DA(1),311,DA,0),"",1)

III. POST-FORM CODE

```

i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

```

2209 SUPER STATEMENT AT RECUR

M/FOM FID # 24

28 STORED: nowhere

29 PREFIELD CODE:

```

s:$d(lvl) lvl=0 i $d(chi(0))>9 s shhfile=1 d fileone`ptfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

```

16 CALLS M/FOM: SUPERVISOR'S STATEMENT (#653)

M/FOM: SUPERVISOR'S STATEMENT (#653)

I. PRE-FORM CODE

```

n file,rptfile,New s:$d(nocomp) file="ptg(iform,99)
+s:$d(nocomp) file="ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s %$d($H),NOT="" d "NOT s DT=Y k X,Y
d sec`fmfm

```

d crinck`mfum

II. WINDOWS

SUPERVISOR'S STATEMENT WINDOW I

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 SUPR STATEMENT AT RECLR	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p('ICC(DA,1),211,DA,0),",",1)		

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,l:filing)
+d shuffle`mfum d noes0`mfum

402 SUPERVISOR'S NAME FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 47

28 STORED: \$p('ICC(DA,402),",",1)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

402.1 SUPERVISOR'S TITLE FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 49

28 STORED: \$p('ICC(DA,402),",",2)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

402.2 SUPERVISOR'S PHONE NUMBER FREE TEXT K:SL(X)>14!(SL(X)<7) X

M/FORM FID # 48

28 STORED: \$p('ICC(DA,402),",",3)
44 POST ID: s X=cm K:SL(X)>14!(SL(X)<7) X s:'\$d(X) cm="" s:cm(" " cm=""

402.4 DATE SUPERVISOR SIGNED DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 50

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p('JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm,"000",9,10) " : " \$e(cm,"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,402),"",5)

WM CONTINUATION OF PAY

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
250 COP REC'D FROM	DATE	S ^DT="E" D ^DT S X=Y R:Y<1 X

M/FORM FID # 91

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",1)

250.5 COP REC'D THRU	DATE	S ^DT="E" D ^DT S X=Y R:Y<1 X
----------------------	------	-------------------------------

M/FORM FID # 93

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",2)

29 PREFIELD CODE: I 'SD(chi(91)) s skip="D"

250.6 COP REC'D GROSS	NUMERIC FORMAT S:X("\$" X=Sp(X,"\$",2) R:X'=X(X'?.N1".2N)!(X>999999.99)!(X<0) X
-----------------------	---

M/FORM FID # 92

28 STORED: Sp(^ICC(DA,250),"",16)

29 PREFIELD CODE: I 'SD(chi(91)) s skip="D"

232 DATE RATE CHANGED	DATE	S ^DT="E" D ^DT S X=Y R:Y<1 X
-----------------------	------	-------------------------------

M/FORM FID # 79

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,232),"",1)

232.1 CHANGED BASE RATE	NUMERIC FORMAT S:X("\$" X=Sp(X,"\$",2) R:X'=X(X'?.N1".2N)!(X>999999.99)!(X<0) X
-------------------------	---

M/FORM FID # 80

28 STORED: \$p(^IOC(DA,232),"",2)

232.2 CHANGED SUBSISTENCE NUMERIC FORMAT S:X "\$" X=\$P(X,"\$",2) K:+X'=X(X'?..NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 84

28 STORED: \$p(^IOC(DA,232),"",3)

232.3 CHANGED QUARTERS NUMERIC FORMAT S:X "\$" X=\$P(X,"\$",2) K:+X'=X(X'?..NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 83

28 STORED: \$p(^IOC(DA,232),"",4)

232.4 CHANGED OTHER RATE NUMERIC FORMAT S:X "\$" X=\$P(X,"\$",2) K:+X'=X(X'?..NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 81

28 STORED: \$p(^IOC(DA,232),"",5)

232.5 CHANGED OTHER SPECIFY FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 82

28 STORED: \$p(^IOC(DA,232),"",6)

29 PREFIELD CODE: I '\$d(cbi(81)) s skip="D"

44 POST ID: s X=on R:SL(X)>30!(SL(X)<1) X s: '\$d(X) on="" s: on "" on=""

2217 WORK INFO DURING COP/COM

M/FORM FID # 27

28 STORED: nowhere

29 PREFIELD CODE:

s: '\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shfile=1 d fileone=ptcfile k shfile
i \$d(file),file,New=0 s filing=\$d(\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr=from s New=0,cbi(0)= "" k file
s lvl=lvl+1
i lvl s DA(0)=DA,CO=CA f i=lvl:-1:1 s: \$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s: \$d(DA(i)) DA(i-1)=DA(i)
s: lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: WORK INFORMATION (#655)

M/FORM: WORK INFORMATION (#655)

I. PRE-FORM CODE

n filen,rptfile,New s: '\$d(nocomp) filen=ptc(iform,99)
+s: \$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),NOT="" d "NOT s DT=Y k X,Y

d sec^fmfm

II. WINDOWS

WORK INFORMATION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DATES & HOURS WORKED	FREE TEXT	K:X[""]!(S(X)-45) X I S(X) K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 1

28 STORED: Sp(^ICC(DA(1),212,DA,0),"",1)
32 POST FILING CODE: d setindx^fmfm
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""
54 INDEX: ^ICC(DA(1),212,"B")
55 LOOKUP NODE: 1^30^Sp(^ICC(DA(1),212,lids,0),"",1)

1 PAY RATE	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
------------	-----------	------------------------

M/FORM FID # 2

28 STORED: Sp(^ICC(DA(1),212,DA,0),"",2)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

2 TOTAL AMOUNT EARNED	NUMERIC FORMAT S:X["\$"] X=SP(X,"\$",2) K:X=X-X(X'?NL".2N)!(X>99999)!(X<0) X
-----------------------	--

M/FORM FID # 3

28 STORED: Sp(^ICC(DA(1),212,DA,0),"",3)

3 TYPE WORK PERFORMED	FREE TEXT	K:SL(X)>20!(SL(X)<1) X
-----------------------	-----------	------------------------

M/FORM FID # 4

28 STORED: Sp(^ICC(DA(1),212,DA,0),"",4)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

4 EMPLOYER NAME	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
-----------------	-----------	------------------------

M/FORM FID # 5

28 STORED: Sp(^ICC(DA(1),212,DA,0),"",5)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

5 EMPLOYER STREET ADDRESS	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
---------------------------	-----------	------------------------

M/FORM FID # 6

28 STORED: Sp(^ICC(DA(1),212,DA,0),"",6)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

6 EMPLOYER CITY/STATE/ZIP	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
---------------------------	-----------	------------------------

M/FORM FID # 7

28 STORED: Sp(IOC(DA(1),212,DA,0),"",7)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=\$e('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0`fmfm

255.6 SICK/ANNUAL LEAVE CLAIMS DRIE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 85

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(IOC(DA,250),"",14)

255.7 SICK/ANNUAL LEAVE CLAIMS DRIE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 86

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(IOC(DA,250),"",15)

29 PREFIELD CODE: I '\$d(chi(85)) s skip="D"

254 CTP CLAIMED FROM DRIE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 89

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(IOC(DA,250),"",10)

254.5 CTP CLAIMED THRU DRIE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 90

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X_"")
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,250),"",11)

29 PREFIELD CODE: I 'SD(cha(89)) s skip="D"

255 COMP CLAIM FROM DATE S NOT="E" D ^NOT S X=Y R:Y<1 X

M/FORM FID # 87

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X_"")
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,250),"",12)

255.5 COMP CLAIMED THRU DATE S NOT="E" D ^NOT S X=Y R:Y<1 X

M/FORM FID # 88

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X_"")
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,250),"",13)

29 PREFIELD CODE: I 'SD(cha(87)) s skip="D"

322 DATE EMP/CLAIMANT SIGNED DATE S NOT="E" D ^NOT S X=Y R:Y<1 X

M/FORM FID # 29

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X_"")
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,303),"",7)

III. POST-FORM CODE

i Sd(file),file,New=0 s:'Sd(lvl) lvl=0
+s filling=Se('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+i upontr fndcm
i file=1!(Sd(ERFLAG)) d OZA T3CPNF, T3OFILE

CA-3 REPORT OF TERMINATION OF DISABILITY AND/OR PAYMENT

PART A--GENERAL

1. NAME OF INJURED EMPLOYEE _____
2. SOCIAL SECURITY NUMBER _____
3. OWCP FILE NUMBER _____
4. DEPARTMENT OR AGENCY _____
5. BUREAU OR OFFICE _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
6. NAME OF REPORTING OFFICE _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
7. DATE AND HOUR OF INJURY _____
8. DATE AND HOUR STOPPED WORK _____
9. DATE AND HOUR PAY STOPPED _____
10. DATE AND HOUR RETURNED TO WORK _____
11. (IF OTHER THAN MONDAY THROUGH FRIDAY)
EMPLOYEE'S WORK WEEK ON RETURN TO DUTY _____
12. PRESENT PAY RATE _____
- A. BASE PAY _____ B. SUBSISTENCE _____ C. QUARTERS _____
- D. OTHER _____ SPECIFY OTHER _____
13. DATES RECEIVED PAY FOR THE FOLLOWING: _____
- A. ANNUAL LEAVE FROM: _____ THROUGH: _____
- B. SICK LEAVE FROM: _____ THROUGH: _____
- C. OTHER LEAVE FROM: _____ THROUGH: _____
- OTHER LEAVE SPECIFY _____
14. HAS WORK ASSIGNMENT BEEN CHANGED? _____
- IF YES, *Press <Return> for DESCRIBE CHANGE DUTY* _____
15. IF INTERRUPTED, SHOW DATES RESUMED _____
- HEALTH BENEFIT _____ OPTIONAL INSURANCE _____
16. IF HEALTH BENEFITS OPTION HAS CHANGED _____
- FEHB NUMBER _____ DATE _____
17. *Press <Return> for REMARKS* _____

PART B--CONTINUATION OF PAY

18. DATE RECEIVED CONTINUATION OF PAY, FROM: _____
- THROUGH: _____
19. SHOW GROSS DOLLAR AMOUNT OF REGULAR PAY _____
20. IF PAY RATE CHANGED, SHOW DATE OF CHANGE _____
21. GIVE NEW RATE _____
- A. BASE PAY _____ B. SUBSISTENCE _____ C. QUARTERS _____
- D. OTHER _____ OTHER SPECIFY _____
22. SIGNATURE OF SUPERVISOR _____
23. TITLE _____
24. DATE _____ PHONE NUMBER _____

*Word Processing Window will appear.

N/FORM: OA-3 (#657)

I. PRE-FORM CODE

```

n filen,rptfile,filing s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1,indexist=2 k ident
s U="" i '$d(DT) s X=$d(SH),%DT="" d "%DT s DT=Y k X,Y
d sec"fmfm
I $D(DA) S:$D("ICC(DA,221)) "ICC(DA,231)~"ICC(DA,222)
I $D(DA) S $P("ICC(DA,221),U,1,3)=""
n filen,rptfile,filing s:'$d(nocomp) filen=ptc(iform,99)

```

II. WINDOWS

WI PART-A

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	POINTER 1004'	EMPLOY(
N/FORM FID # 36		
17 EXTERNAL DEFAULT: NAME		
18 INTERNAL DEFAULT: NAMEPTR		
20 CONVERSION CODE: s am=6e(am=""',('(\$d("EMPLOY(am,0))#10):am,1:\$p("EMPLC/(am,0),"",1))		
28 STORED: \$p("ICC(DA,0),"",3)		
32 POST FILING CODE: d setindx"fmfm		
44 POST ID: i \$d(new) s:'\$d("ICC("C",new,x1)) msg="CLAIM NOT LOGGED!",y=24,am=""		
54 INDEX: "ICC("C")		
55 LOOKUP NODE:		
36"30"\$p("ICC(Lids,0),"",3)		
7"23"\$p("ICC(Lids,0),"",2)		
3 SEX	FREE TEXT	K:\$L(X)>9!(\$L(X)<9)!'(XQ9N) X
N/FORM FID # 2		
28 STORED: \$p("ICC(DA,0),"",4)		
44 POST ID: s X=am K:\$L(X)>9!(\$L(X)<9)!'(XQ9N) X s:'\$d(X) am="" s:am["" am=""		
137 OACP FILE NUMBER	FREE TEXT	K:\$L(X)>13!(\$L(X)<1) X
N/FORM FID # 3		
28 STORED: \$p("ICC(DA,130),"",6)		
44 POST ID: s X=am K:\$L(X)>13!(\$L(X)<1) X s:'\$d(X) am="" s:am["" am=""		
11 AGENCY	POINTER 1074'	AGENCY(0,
N/FORM FID # 4		

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d(^AGENCY(0,cm,0))#10):cm,1:\$p(^AGENCY(0,cm,0),"",2))
28 STORED: \$p(^IOC(DA,2),"",1)

13 DUTY STATION POINTIER 1062' INJ(1062,

M/FORM FID # 22

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d(^INJ(1062,cm,0))#10):cm,1:\$p(^INJ(1062,cm,0),"",1))
28 STORED: \$p(^IOC(DA,9),"",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 23

28 STORED: \$p(^IOC(DA,9),"",5)
44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

13.5 DUTY STATION CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 24

28 STORED: \$p(^IOC(DA,9),"",6)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

13.6 DUTY STATION STATE POINTIER 1003' DIZ(1003,

M/FORM FID # 25

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^IOC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 26

28 STORED: \$p(^IOC(DA,9),"",8)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s: '\$d(X) cm="" s:cm("" cm=""

14 REPORTING OFFICE POINTIER 1063' INJ(1063,

M/FORM FID # 27

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d(^INJ(1063,cm,0))#10):cm,1:\$p(^INJ(1063,cm,0),"",1))
28 STORED: \$p(^IOC(DA,10),"",1)

14.01 RO ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 28

28 STORED: \$p(^IOC(DA,10),"",2)
44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s: '\$d(X) cm="" s:cm("" cm=""

14.02 RO CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 29

28 STORED: \$p(^ICC(DA,10),"",3)

44 POST ID: s %cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm{"" cm=""

14.03 FO STATE

FOINTER 1003' DIZ(1003,

M/FORM FID # 30

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,10),"",4)

14.04 FO ZIP CODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 31

28 STORED: \$p(^ICC(DA,10),"",5)

44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm{"" cm=""

1 INJURY/ILLNESS DATE & TI

DATE

S %DT="ET" D %DT S %Y K:Y<1 X I \$D(X) K:\$P(X,"",1)>DT X

M/FORM FID # 7

20 CONVERSION CODE:

s %="(" s:+\$s(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$s(cm,4,5)) " " s:+\$s(cm,6,7) %K \$s(cm,6,7) ", " s %K (1700+\$s(cm,1,3))
s:\$p(cm,".",2)'="" %X " at " \$s(cm "000",9,10) _:" \$s(cm "000",11,12) s %X_"
s:X="(1700)" %="" s cm=X

28 STORED: \$p(^ICC(DA,0),"",2)

32 POST FILING CODE: d setindx fmfcm

54 INDEX: ^ICC("D")

55 LOOKUP NODE:

7^23^\$p(^ICC(Lids,0),"",2)

36^30^\$p(^ICC(Lids,0),"",3)

229 D/T STOPPED WORK

DATE

S %DT="ET" D %DT S %Y K:Y<1 X

M/FORM FID # 8

20 CONVERSION CODE:

s %="(" s:+\$s(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$s(cm,4,5)) " " s:+\$s(cm,6,7) %K \$s(cm,6,7) ", " s %K (1700+\$s(cm,1,3))
s:\$p(cm,".",2)'="" %X " at " \$s(cm "000",9,10) _:" \$s(cm "000",11,12) s %X_"
s:X="(1700)" %="" s cm=X

28 STORED: \$p(^ICC(DA,221),"",1)

230 D/T PAY STOPPED

DATE

S %DT="ET" D %DT S %Y K:Y<1 X

M/FORM FID # 9

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,221),""",2)

231 D/T RETURNED TO WORK DATE S %DT="ET" D %DT S X=Y K:Y<1 X

M/FORM FID # 10

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,221),""",3)

324 NEW WORK WEEK FREE TEXT K:SL(X)>30!(SL(X)<3) X

M/FORM FID # 11

28 STORED: Sp("ICC(DA,303),""",8)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm[""" cm=""

231.1 RETURNED TO WORK BASE RA NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 56

28 STORED: Sp("ICC(DA,231),""",1)

231.3 RETURN TO WORK SUBSIDEN NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 39

28 STORED: Sp("ICC(DA,231),""",3)

231.4 RETURN TO WORK QUARTERS NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 38

28 STORED: Sp("ICC(DA,231),""",5)

231.5 RETURN TO WORK OTHER RAT NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 37

28 STORED: Sp("ICC(DA,231),""",7)

231.6 RETURN TO WORK SPECIFY 0 FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 40

28 STORED: Sp(ICC(DA,231),"",9)
 29 PREFIELD CODE: s:\$d(cbi(37)) skip="D"
 44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:\$d(X) cm="" s:cm "" cm=""

W2 PART-A

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
253 ANNUAL LEAVE FROM	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
M/FORM FID # 49		
20 CONVERSION CODE: s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) _:" Se(cm "000",11,12) s X=X _)" s:X="(1700)" X="" s cm=X		
28 STORED: Sp(IGL(DA,250),"",5)		
253.5 ANNUAL LEAVE THRU	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
M/FORM FID # 50		
20 CONVERSION CODE: s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) _:" Se(cm "000",11,12) s X=X _)" s:X="(1700)" X="" s cm=X		
28 STORED: Sp(ICC(DA,250),"",6)		
252 SICK LEAVE FROM	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
M/FORM FID # 51		
20 CONVERSION CODE: s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) _:" Se(cm "000",11,12) s X=X _)" s:X="(1700)" X="" s cm=X		
28 STORED: Sp(ICC(DA,250),"",3)		
252.5 SICK LEAVE THRU	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
M/FORM FID # 52		
20 CONVERSION CODE: s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) _:" Se(cm "000",11,12) s X=X _)" s:X="(1700)" X="" s cm=X		

28 STORED: \$p(^TCC(DA,250),"",4)

251 OTHER LEAVE FROM DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 53

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10)_" ":"\$e(cm,"000",11,12) s X=X_"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^TCC(DA,250),"",7)

251.5 OTHER LEAVE THRU DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 55

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10)_" ":"\$e(cm,"000",11,12) s X=X_"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^TCC(DA,250),"",8)

251.6 OTHER LEAVE SPECIFY FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 54

28 STORED: \$p(^TCC(DA,250),"",9)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

329 DUTY CHANGED DUE TO DISA SET OF CODES Y:YES
N:NO

M/FORM FID # 14

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",":",2),":",1),cm=\$p(cm,":",1)

28 STORED: \$p(^TCC(DA,303),"",9)

329.1 DESCRIBE CHANGED DUTY

M/FORM FID # 21

28 STORED: nowhere

29 PREFIELD CODE:

I '\$d(cbi(14)) s skip="D",cbi(0,21)=""
i \$d(cbi(14)) s:cbi(14){"N"i(cbi(14)){"n" skip="D",cbi(0,21)=""
s:\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shfile=l d filecm ptrfile k shfile
i \$d(file),file,New=0 s filing:\$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr'fscrm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0-DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

16 CALLS M/FORM: DESCRIBE CHANGED DUTY (#661)

M/FORM: DESCRIBE CHANGED DUTY (#661)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),%DT="" d ~NOT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm

II. WINDOWS

DESCRIBE CHANGED DUTY

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DESCRIBE CHANGED DUTY	WORD PROCESSING	

M/FORM FID # 1

28 STORED: Sp('IOC(DA(1),314,DA(0),",",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$e('\$d(filing):lvl,lvb)filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

330 DATE HEALTH DEDUCTION RE DATE S NOT="E" D ~NOT S X=Y K:Y<1 X

M/FORM FID # 15

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+Se(cm,4,5)) " ' s:+Se(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) "-" \$e(cm "000",11,12) s X=X "
s:X="(1700)" X="" s cm=X

28 STORED: Sp('IOC(DA,303),",",10)

331 DATE OPT INS DEDUCTION R DATE S NOT="E" D ~NOT S X=Y K:Y<1 X

M/FORM FID # 16

20 CONVERSION CODE:

```
s X= (" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X
```

28 STORED: Sp("IOC(DA,303),""",11)

25 FEBB CODE

FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 17

28 STORED: Sp("IOC(DA,3),""",13)

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

333 NEW HEALTH PLAN DATE

DATE

S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 18

20 CONVERSION CODE:

```
s X= (" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X
```

28 STORED: Sp("IOC(DA,303),""",13)

54 REMARKS

M/FORM FID # 19

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(dbi(0))>9 s shhfile=l d fileone=ptrfilel k shhfile
i $d(file),file,New=0 s filling=Se('$d(filing):lvl+1,lvl'<filling:lvl+1,1:filling)
+d upcntr=fnfcm s New=0,dbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=CA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CMIS M/FORM: REMARKS (#606)

M/FORM: REMARKS (#606)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=ptrc(iform,99)
+s:$d(nocomp) filen=ptrc(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),NOT="" d NOT s DT=Y k X,Y
d sec=fnfcm
```

d crinck`fmfm

II. WINDOWS

REMARKS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 REMARKS	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: Sp(`ICC(DA(1),8,DA,0),"" ,1)		

III. POST-FORM CODE

i Sd(file),file s filing=Se('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

W3 COP

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
250 COP RECVD FROM	DATE	S NOT="E" D `NOT S X=Y R:Y<1 X
M/FORM FID # 41		
20 CONVERSION CODE: s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")" s:X="(1700)" X="" s cm=X		
28 STORED: Sp(`ICC(DA,250),"" ,1)		
250.5 COP RECVD THRU	DATE	S NOT="E" D `NOT S X=Y R:Y<1 X
M/FORM FID # 43		
20 CONVERSION CODE: s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",", +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'="" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")" s:X="(1700)" X="" s cm=X		
28 STORED: Sp(`ICC(DA,250),"" ,2)		
29 PREFIELD CODE: s:'Sd(cbi(41)) skip="D"		
250.6 COP RECVD GROSS	NUMERIC FORMAT	S:X("S" X=SP(X,"S",2) R:X'=>X(X'?.N1"."2N):(X>999999.99)1(X<0) X

M/FORM FID # 42

28 STORED: Sp("IOC(DA,250),"",16)

29 PREFIELD CODE: s:'\$d(cbl(41)) skip="D"

232 DIME PRICE CHANGED DIME S NOT="E" D "NOT S X=Y K:Y<1 X

M/FORM FID # 57

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X "_"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,232),"",1)

232.1 CHANGED BASE PRICE NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'-X(X'?.NL"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 44

28 STORED: Sp("IOC(DA,232),"",2)

29 PREFIELD CODE: s:'\$d(cbl(57)) skip="D"

232.2 CHANGED SUBSTANCE NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'-X(X'?.NL"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 48

28 STORED: Sp("IOC(DA,232),"",3)

29 PREFIELD CODE: s:'\$d(cbl(57)) skip="D"

232.3 CHANGED QUARTERS NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'-X(X'?.NL"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 47

28 STORED: Sp("IOC(DA,232),"",4)

29 PREFIELD CODE: s:'\$d(cbl(57)) skip="D"

232.4 CHANGED OTHER PRICE NUMERIC FORMAT S:X["\$" X=Sp(X,"\$",2) K:+X'-X(X'?.NL"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 45

28 STORED: Sp("IOC(DA,232),"",5)

29 PREFIELD CODE: s:'\$d(cbl(57)) skip="D"

232.5 CHANGED OTHER SPECIFY FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 46

28 STORED: Sp("IOC(DA,232),"",6)

29 PREFIELD CODE: s:'\$d(cbl(45)) skip="D"

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

402 SUPERVISOR'S NAME FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 34

28 STORED: Sp(IOC(DA,402),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

402.1 SUPERVISOR'S TITLE FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 33

28 STORED: Sp(IOC(DA,402),"",2)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

402.2 SUPERVISOR'S PHONE NUMBER FREE TEXT K:SL(X)>14!(SL(X)<7) X

M/FORM FID # 32

28 STORED: Sp(IOC(DA,402),"",3)

44 POST ID: s X=cm K:SL(X)>14!(SL(X)<7) X s:'\$d(X) cm="" s:cm("" cm=""

402.4 DATE SUPERVISOR SIGNED DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 35

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB..MAR..APR..MAY..JUN..JUL..AUG..SEP..OCT..NOV..DEC",",",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",Z)'="" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(IOC(DA,402),"",5)

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing=6s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upontr fmfcm
i file=1!(\$d(BFFLAG)) d O3 T3CPNF, T3CPFILE

CA-5 CLAIM FOR COMPENSATION BY WIDOW, WIDOWER

1. NAME OF DECEASED EMPLOYEE _____
2. DATE OF BIRTH (MO., DAY, YEAR) _____
3. DATE OF INJURY (MO., DAY, YEAR) _____
4. DATE OF DEATH (MO., DAY, YEAR) _____
5. SOCIAL SECURITY NUMBER _____
6. NAME OF EMPLOYING AGENCY _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
7. NATURE OF INJURY WHICH CAUSED DEATH _____
Press <Return> for NATURE OF INJURY/ILLNESS

CLAIM OF SURVIVING HUSBAND OR WIFE (49-#13)

8. NAME _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
9. YOUR DATE OF BIRTH (MO., DAY, YEAR) _____
10. DATE OF MARRIAGE TO EMPLOYEE _____
11. WERE YOU LIVING WITH EMP AT TIME DEATH? _____
12. WERE YOU MARRIED TO OTHER THAN EMPLOYEE? _____
13. WAS EMP MARRIED TO OTHER THAN YOURSELF? _____

CLAIM COMPENSATION BY WIDOW, WIDOWER (PAGE 2)

14. EMP'S CHILDREN WHO ENTITLED TO COMP
Press <Return> for CHILDREN OF THIS MARRIAGE
15. IF LEGAL GUARDIAN APPOINTED FOR CHILD,
Press <Return> for CHILD & LEGAL GUARDIAN
16. LIST OTHER RELATIVES WHO FULLY DEPENDENT
Press <Return> for OTHER DEPENDENTS
17. IF EMPLOYEE WAS EVER IN THE ARMED FORCES OF THE UNITED STATES,
GIVE:
SERVICE NUMBER: _____
BRANCH OF SERVICE: _____
PERIOD OF SERVICE: _____
18. IF APPLICATION HAS BEEN MADE FOR VETERANS ADMINISTRATION (VA)
BENEFITS BECAUSE OF EMPLOYEE'S DEATH, GIVE:
VA CLAIM NUMBER: _____
ADDRESS OF VA OFFICE WHERE CLAIM IS FILED: _____
CITY _____ STATE _____ ZIP CODE _____

CA-5 CLAIM FOR COMPENSATION BY WIDOW, WIDOWER (PAGE 3)

19. IF APPLICATION HAS BEEN MADE OF U.S. CIVIL SERVICE ANNUITY BECAUSE OF EMPLOYEE'S DEATH, GIVE:
CSF CLAIM NUMBER: _____
DATE ANNUITY BEGAN: _____
AMOUNT PAID PER MONTH: \$ _____
20. IF A CLAIM HAS BEEN MADE AGAINST A 3RD PARTY
Press <Return> for NAME ADDRESS THIRD PARTY
21. TOTAL BURIAL EXPENSE \$ _____
22. AMOUNT OF BURIAL EXPENSE PAYABLE BY VA \$ _____
23. NAME OF PARTY WHOSE PAID BURIAL EXPENSE _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
25. ADDRESS OF PERSON FILING CLAIM _____
CITY _____ STATE _____ ZIP CODE _____
26. DATE (MO., DAY, YEAR) _____

ATTENDING PHYSICIAN'S REPORT

1. NAME OF DESEASED EMPLOYEE (SEE #1 OF PAGE 1) _____
2. DATE OF DEATH (SEE #4 OF PAGE 1) _____
3. HISTORY OF INJURY RELATED TO DISEASE
Press <Return> for DOC'S NARRATIVE OF INJURY
4. IF TREATED FOR DISEASE, GIVE DIAGNOSIS:
Press <Return> for DOC'S TREATMENT/DIAGNOSIS
5. IF DEATH WAS NOT INSTANTANEOUS,
Press <Return> for TREATMENT PRIOR TO DEATH
6. SHOW DATES ON WHICH TREATMENT WAS GIVEN _____
7. *Press <Return> for DIRECT CAUSE OF DEATH*
8. *Press <Return> for CONTRIBUTORY CAUSE OF DEATH*
9. WAS THE EMP'S DEATH DUE TO THE REPORT #3?
GIVE MEDICAL REASONS AND CAUSAL RELATIONSHIP
Press <Return> for EXPLAIN CAUSAL RELATIONSHIP
10. WAS A BIOPSY OR AUTOPSY PERFORMED? _____
IF YES, GIVE NAME OF PHYSICIAN _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
11. NAME _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

CA-5 CON'T

CHILDREN OF THIS MARRIAGE

NAME _____
RELATIONSHIP _____ DATE OF BIRTH _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

CHILD & LEGAL GUARDIAN

CHILD NAME _____
GUARDIAN NAME _____
GUARDIAN'S ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

OTHER DEPENDENTS

NAME _____
RELATIONSHIP _____ DATE OF BIRTH _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

THIRD PARTY INFORMATION

NAME _____ SETTLEMENT AMOUNT _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

*Word Processing Window will appear.

M/FORM: CA-5 (#672)

I. PRE-FORM CODE

```
n filen,rptfile,filing s:'$d(nocomp) filen~ptc(ifcom,99)
+s:$d(nocomp) filen~ptg(903,ifcom,99) s New=0,rptfile=1,idecist=2 k ident
s U~"" i '$d(DT) s %~$d($H),%~="" d ~DT s DT=Y k X,Y
d sec~fufm
```

II. WINDOWS

CAS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	POINTER 1004'	EMPLOY(
--------	---------------	---------

M/FORM FID # 81

```
17 EXTERNAL DEFAULT: NAME
18 INTERNAL DEFAULT: NAMEPTR
20 CONVERSION CODE: s cm=$e(cm~""~""~"('$d('EMPLOY(cm,0))#10):cm,1:$p('EMPLOY(cm,0),""~",1))
28 STORED: $p('ICC(DA,0),""~",3)
32 POST FILING CODE: d setindx~fufm
44 POST ID: I SD(new) s:'$d('ICL("C",new,x1)) s cm~""~",y=24,msg="CLAIM NOT LOGGED!"
54 INDEX: 'ICC("C")
55 LOOKUP NODE:
81~20~$p('ICC(lids,0),""~",3)
3~23~$p('ICC(lids,0),""~",2)
```

6 BIRTHDATE	DATE	S NOT="E" D ~DT S X=Y K:Y<1 X
-------------	------	-------------------------------

M/FORM FID # 2

```
20 CONVERSION CODE:
s X~(" s:+$e(cm,4,5) X~X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X~X $e(cm,6,7) ", " s X~X (1700+$e(cm,1,3))
s:$p(cm,".",2)~"" X~X " at " $e(cm~"000",9,10) ~:" $e(cm~"000",11,12) s X~X ~")"
s:X~"(1700)" X~"" s cm~X
28 STORED: $p('ICC(DA,0),""~",7)
```

1 INJURY/ILLNESS DATE & TI	DATE	S NOT="E" D ~DT S X=Y K:Y<1 X I SD(X) K:\$P(X,".",1)>DT X
----------------------------	------	---

M/FORM FID # 3

```
20 CONVERSION CODE:
s X~(" s:+$e(cm,4,5) X~X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X~X $e(cm,6,7) ", " s X~X (1700+$e(cm,1,3))
s:$p(cm,".",2)~"" X~X " at " $e(cm~"000",9,10) ~:" $e(cm~"000",11,12) s X~X ~")"
s:X~"(1700)" X~"" s cm~X
28 STORED: $p('ICC(DA,0),""~",2)
32 POST FILING CODE: d setindx~fufm
```

54 INDEX: "ICC("D")
 55 LOOKUP CODE:
 3"23"Sp("ICC(Lids,0),"",2)
 81"20"Sp("ICC(Lids,0),"",3)

103 DATE OF DEPTH DATE S "DT="ET" D "DT S X-Y K:Y<1 X I \$D(X) K:X\$P("ICC(DA,0),U,2) X

M/FORM FID # 40

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","",
 +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,"",2)"=" X=X " at " Se(cm"000",9,10)"=" Se(cm"000",11,12) s X=X "
 s:X="(1700)" X=" s cm=X

28 STORED: Sp("ICC(DA,100),"",4)

3 SSN FREE TEXT K:SL(X)>9!(SL(X)<9)!"(XQ9N) X

M/FORM FID # 5

28 STORED: Sp("ICC(DA,0),"",4)

44 POST ID: s X=cm K:SL(X)>9!(SL(X)<9)!"(XQ9N) X s:'\$d(X) cm="" s:cm(""" cm=""

13 DUTY STATION FOINIER 1062' INU(1062,

M/FORM FID # 41

20 CONVERSION CODE: s cm=Se(cm=""":""',(\$d("INU(1062,cm,0))#10):cm,1:Sp("INU(1062,cm,0),"",1))

28 STORED: Sp("ICC(DA,9),"",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 42

28 STORED: Sp("ICC(DA,9),"",5)

44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

13.5 DUTY STATION CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 43

28 STORED: Sp("ICC(DA,9),"",6)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

13.6 DUTY STATION STATE FOINIER 1003' DIZ(1003,

M/FORM FID # 44

20 CONVERSION CODE: s cm=Se(cm=""":""',(\$d("DIZ(1003,cm,0))#10):cm,1:Sp("DIZ(1003,cm,0),"",1))

28 STORED: Sp("ICC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 45

28 STORED: Sp(^ICC(DA,9), "", 8)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""] cm=""

101.1 NATURE OF INJURY EXPLAN

M/FORM FID # 7

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbl(0))>9 s shhfile=1 d fileone^ptrfile: k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fufm s New=0,cbl(0)="" k file
s lvl=lvl-1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: NATURE OF INJURY/ILLNESS (#642)

M/FORM: NATURE OF INJURY/ILLNESS (#642)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=^ptc(iform,99)
+s:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),WDT="" d ^WDT s DT=Y k X,Y
d sec^fufm
d crinck^fufm
```

II. WINDOWS

NATURE OF INJURY/ILLNESS Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NATURE OF INJURY EXPLAN	WORD PROCESSING	

M/FORM FID # 1

28 STORED: Sp(^ICC(DA(1),101,DA,0), "", 1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle^fufm d node0^fufm
```

CAS window 2

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

502 SPOUSE AT DEATH	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
---------------------	-----------	------------------------

M/FORM FID # 56

28 STORED: Sp(IOC(DA,501),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

502.1 ADDRESS OF SPOUSE	FREE TEXT	K:SL(X)>55!(SL(X)<1) X
-------------------------	-----------	------------------------

M/FORM FID # 57

28 STORED: Sp(IOC(DA,501),"",2)

44 POST ID: s X=cm K:SL(X)>55!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

502.2 CITY OF SPOUSE	FREE TEXT	K:SL(X)>20!(SL(X)<1) X
----------------------	-----------	------------------------

M/FORM FID # 58

28 STORED: Sp(IOC(DA,501),"",3)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

502.3 STATE OF SPOUSE	POINTER 1003' DIZ(1003,
-----------------------	-------------------------

M/FORM FID # 59

20 CONVERSION CODE: s cm=\$s(cm:"","'(\$d(DIZ(1003,cm,0))#10):cm,1:Sp(DIZ(1003,cm,0),"",1))

28 STORED: Sp(IOC(DA,501),"",4)

502.4 ZIPCODE OF SPOUSE	FREE TEXT	K:SL(X)>10!(SL(X)<5) X
-------------------------	-----------	------------------------

M/FORM FID # 60

28 STORED: Sp(IOC(DA,501),"",5)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""] cm=""

530 SPOUSE DOB	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
----------------	------	-------------------------------

M/FORM FID # 8

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=K \$e(cm,6,7) " " s X=K (1700+\$e(cm,1,3))
s:Sp(cm,"",2)"-" X=K " at " \$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=K ")"
s:X="(1700)" X="" s cm=K

28 STORED: Sp(IOC(DA,501),"",6)

531 DATE OF MARRIAGE	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
----------------------	------	-------------------------------

M/FORM FID # 9

20 CONVERSION CODE:

s %="(" s:+\$(cm,4,5) %K \$("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ". "
 ,+\$(cm,4,5)) " " s:+\$(cm,6,7) %K \$(cm,6,7) " " s %K (1700+\$(cm,1,3))
 s:\$(cm,".",2)'="" %K " at " \$(cm"000",9,10) _:" \$(cm"000",11,12) s %K _"
 s:%="(1700)" %="" s cm%K

28 STORED: \$("ICC(DA,515),"",1)

532 LIVING W/SPOUSE AT DEATH SET OF CODES Y:YES
 N:NO

M/FORM FID # 10

20 CONVERSION CODE: s cm=\$(\$("Y:YES;N:NO;" _cm _:"2)," _:"1),cm=\$(cm," _:"1)
 28 STORED: \$("ICC(DA,515),"",2)

533 SPOUSE PREVIOUSLY MARRI SET OF CODES Y:YES
 N:NO

M/FORM FID # 11

20 CONVERSION CODE: s cm=\$(\$("Y:YES;N:NO;" _cm _:"2)," _:"1),cm=\$(cm," _:"1)
 28 STORED: \$("ICC(DA,515),"",3)

534 EMP PREVIOUSLY MARRIED SET OF CODES Y:YES
 N:NO

M/FORM FID # 12

20 CONVERSION CODE: s cm=\$(\$("Y:YES;N:NO;" _cm _:"2)," _:"1),cm=\$(cm," _:"1)
 28 STORED: \$("ICC(DA,515),"",4)

CAS Window 3

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

535 CHILDREN THIS MARRIAGE

M/FORM FID # 13

28 STORED: nowhere

29 PREFIELD CODE:

s:\$(lvl) lvl=0 i \$(cbl(0))>9 s shhfile=l d filecne`ptrfilel k shhfile
 i \$(file),file,New=0 s filling=\$(\$(filling):lvl+1,lvl'<filling:lvl+1,1:filling)
 +d upcntr`snfm s New=0,cbl(0)="" k file
 s lvl=lvl+1
 i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:\$(DA(i-1)) DA(i)=CA(i-1)
 k DA(0) S DA="",DI=""

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s:\$(DA(i)) DA(i-1)=CA(i)
 s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: CHILDREN OF THIS MARRIAGE (#673)

M/FORM: CHILDREN OF THIS MARRIAGE (#673)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(rocomp) filen=ptc(iform,99)
+s:\$d(rocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U=" " i '\$d(DT) s X=\$d(SH),NOT=" " d "NOT s DT=Y k X,Y
d sec'nfim

II. WINDOWS

CHILDREN OF THIS MARRIAGE Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NAME	FREE TEXT	K:SL(X)>30!(SL(X)<3) X

M/FORM FID # 1

28 STORED: \$p('TOC(DA(1),516,DA,0),"",1)
32 POST FILING CODE: d setindx'nfim
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm[""] cm=""
54 INDEX: TOC(DA(1),516,"B")
55 LOOKUP NODE: 1"30"\$p('TOC(DA(1),516,1ids,0),"",1)

1 RELATIONSHIP	FREE TEXT	K:SL(X)>30!(SL(X)<3) X
----------------	-----------	------------------------

M/FORM FID # 2

28 STORED: \$p('TOC(DA(1),516,DA,0),"",2)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm[""] cm=""

2 DATE OF BIRTH	DATE	S NOT="E" D "NOT S X=Y K:Y<1 X
-----------------	------	--------------------------------

M/FORM FID # 3

20 CONVERSION CODE:
s X=" " s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","")
+Se(cm,4,5)) " " s:+\$e(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:\$p(cm,".",2)" " X=X " at " Se(cm"000",9,10) " " Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p('TOC(DA(1),516,DA,0),"",3)

3 ADDRESS	FREE TEXT	K:SL(X)>60!(SL(X)<1) X
-----------	-----------	------------------------

M/FORM FID # 5

28 STORED: \$p('TOC(DA(1),516,DA,0),"",4)
44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

4 CITY	FREE TEXT	K:SL(X)>23!(SL(X)<1) X
--------	-----------	------------------------

M/FORM FID # 6

28 STORED: \$p('ICC(DA(1),516,DA,0),"",5)
 44 POST ID: s %cm K:\$L(X)>231(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

5 STATE POINTER 1003' DIZ(1003,

M/FORM FID # 9

20 CONVERSION CODE: s cm=\$s(cm="" ":" '(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))
 22 STORED: \$p('ICC(DA(1),516,DA,0),"",6)

6 ZIP CODE FREE TEXT K:\$L(X)>101(\$L(X)<5) X

M/FORM FID # 8

28 STORED: \$p('ICC(DA(1),516,DA,0),"",7)
 44 POST ID: s %cm K:\$L(X)>101(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=\$s(' \$d(filing):lvl,lvl>filing:lvl,1:filing)
 +d node0`fmfm

537 CHILD & LEGAL GUARDIAN

M/FORM FID # 15

28 STORED: nowhere
 29 PREFIELD CODE:
 s:'\$d(lvl) lvl=0 i \$d(obi(0))>9 s shhfile=1 d fileone`ptrfilel k shhfile
 i \$d(file),file,New=0 s filing=\$s(' \$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
 +d upontr`fmfm s New=0,obi(0)="" k file
 s lvl=lvl+1
 i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
 k DA(0) S DA="",D1=""
 30 POSTFIELD CODE:
 i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
 s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

16 CALLS M/FORM: CHILD & LEGAL GUARDIAN (#675)

M/FORM: CHILD & LEGAL GUARDIAN (#675)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=`ptc(iform,99)
 +s:\$d(nocomp) filen=`ptg(903,iform,99) s New=0,rptfile=1
 s U="" i '\$d(UT) s %\$ad(\$h),NOT="" d `NOT s DT=Y k X,Y
 d sec`fmfm

II. WINDOWS

CHILD & LEGAL GUARDIAN Window 1

DATA ELEMENT	DATA TYPE	FOUNTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 CHILD NAME	FREE TEXT	K:SL(X)>30!(SL(X)<3) X

M/FORM FID # 1

28 STORED: Sp('TCC(DA(1),518,DA,0),'',1)
 32 POST FILING CODE: d setindk'fufm
 44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm["" cm=""
 54 INDEX: 'TCC(DA(1),518,"B")
 55 LOOKUP NOTE: 1'30'Sp('TCC(DA(1),518,1ids,0),'',1)

1 GUARDIAN NAME	FREE TEXT	K:SL(X)>30!(SL(X)<3) X
-----------------	-----------	------------------------

M/FORM FID # 2

28 STORED: Sp('TCC(DA(1),518,DA,0),'',2)
 44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm["" cm=""

2 GUARDIAN'S ADDRESS	FREE TEXT	K:SL(X)>60!(SL(X)<1) X
----------------------	-----------	------------------------

M/FORM FID # 3

28 STORED: Sp('TCC(DA(1),518,DA,0),'',3)
 44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

3 CITY	FREE TEXT	K:SL(X)>23!(SL(X)<1) X
--------	-----------	------------------------

M/FORM FID # 4

28 STORED: Sp('TCC(DA(1),518,DA,0),'',4)
 44 POST ID: s X=cm K:SL(X)>23!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

4 STREET	FOUNTER 1003' DIEZ(1003,
----------	--------------------------

M/FORM FID # 7

20 CONVERSION CODE: s cm=\$s(cm=""',('Sd('DIEZ(1003,cm,0))#10):cm,1:Sp('DIEZ(1003,cm,0),'',1))
 28 STORED: Sp('TCC(DA(1),518,DA,0),'',5)

5 ZIP CODE	FREE TEXT	K:SL(X)>10!(SL(X)<5) X
------------	-----------	------------------------

M/FORM FID # 6

28 STORED: Sp('TCC(DA(1),518,DA,0),'',6)
 44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
 +d node0'fufm

538 OTHER DEPENDENTS

M/FORM FID # 39

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone^ptrfile1 k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:ziling)
+d upontr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: OTHER DEPENDENTS (#685)

M/FORM: OTHER DEPENDENTS (#685)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=^ptc(iform,99)
+e:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),NOT="" d ^NOT s DT=Y k X,Y
d sec^fmfm
```

II. WINDOWS

OTHER DEPENDENTS Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<3) X

M/FORM FID # 1

```
28 STORED: $p(^IOC(DA(1),519,DA(0),"",1)
32 POST FILING CODE: d setindx^fmfm
44 POST ID: s X=cm K:$L(X)>30!($L(X)<3) X s:'$d(X) cm="" s:cm(""" cm=""
54 INDEX: ^IOC(DA(1),519,"B")
55 LOOKUP NODE: 1^30^$p(^IOC(DA(1),519,lids,0),"",1)
```

1 RELATIONSHIP	FREE TEXT	K:\$L(X)>30!(\$L(X)<3) X
----------------	-----------	--------------------------

M/FORM FID # 2

```
28 STORED: $p(^IOC(DA(1),519,DA(0),"",2)
44 POST ID: s X=cm K:$L(X)>30!($L(X)<3) X s:'$d(X) cm="" s:cm(""" cm=""
```

2 DATE OF BIRTH	DATE	S NOT="E" D ^NOT S X=Y K:Y<1 X
-----------------	------	--------------------------------

M/FORM FID # 3

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","")
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,"",2)"="" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA(1),519,DA,0),"",3)

3 ADDRESS FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 4

28 STORED: Sp("IOC(DA(1),519,DA,0),"",4)

44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

4 CITY FREE TEXT K:SL(X)>23!(SL(X)<1) X

M/FORM FID # 5

28 STORED: Sp("IOC(DA(1),519,DA,0),"",5)

44 POST ID: s X=cm K:SL(X)>23!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

5 STATE POINTER 1003' DIZ(1003,

M/FORM FID # 8

20 CONVERSION CODE: s cm=Se(cm=""":""',('\$d('DIZ(1003,cm,0))#10):cm,1:Sp('DIZ(1003,cm,0),"",1))

28 STORED: Sp("IOC(DA(1),519,DA,0),"",6)

6 ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 7

28 STORED: Sp("IOC(DA(1),519,DA,0),"",7)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=Se('\$d(filing):lvl,lvl>filing:lvl,1:filing)
 +d node0" fofom

505 SERVICE NO FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 16

28 STORED: Sp("IOC(DA,500),"",3)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

506 BRANCH OF SERVICE

SET OF CODES

A:ARMY

N:NAVY

M:MARINES

F:AIR FORCE

C:COAST GUARD

M/FORM FID # 17

28 STORED: \$p(^ICC(DA,500),"",4)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

507 PERIOD OF SERVICE FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 18

28 STORED: \$p(^ICC(DA,500),"",5)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

710 VA CLAIM NUMBER FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 46

28 STORED: \$p(^ICC(DA,701),"",1)

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

710.2 VA STREET ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 76

28 STORED: \$p(^ICC(DA,701),"",3)

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

710.3 VA CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 49

28 STORED: \$p(^ICC(DA,701),"",4)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

710.4 VA STREET POINTER 1003' DIEZ(1003,

M/FORM FID # 50

20 CONVERSION CODE: s cm=Se(cm="" : "" ,('\$d(^DIEZ(1003,cm,0))#10):cm,1:\$p(^DIEZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,701),"",5)

710.5 VA ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 51

28 STORED: \$p(^ICC(DA,701),"",6)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

CAS Window 4

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
714 OTHER ANNUITIES CLAIM NU	FREE TEXT	K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 52

28 STORED: Sp('ICC(DA,707),'",1)
 44 POST ID: s %cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm ""

714.1 OTHER ANNUITY DRIE BEGIN DATE S NOT="E" D NOT S %Y X:Y<1 X

M/FORM FID # 53

20 CONVERSION CODE:

s %="(" s:+Se(cm,4,5) %K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) %K Se(cm,6,7) ", " s %K (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'="" %K " at " Se(cm"000",9,10)'": Se(cm"000",11,12) s %K "
 s:%="(1700)" %="" s cmK

28 STORED: Sp('ICC(DA,707),'",2)

714.2 OTHER ANNUITY AMOUNT NUMERIC FORMAT S:X "\$" %SP(X,"\$",2) K:+K'-%(X'7.N1"."2N)!(X>99999999.99)!(X<0) X

M/FORM FID # 54

28 STORED: Sp('ICC(DA,707),'",3)

141.1 THIRD PARTY

M/FORM FID # 21

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(dtl(0))>9 s sthfile=l d filen="ptcfile k sthfile
 i \$d(file),file,New=0 s filing=Se('\$(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
 +d upcntr" fctm s New=0,dtl(0)="" k file
 s lvl=lvl+1
 i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
 k DA(0) S DA="" ,DL=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
 s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

16 CALLS M/FORM: NAME/ADDRESS THIRD PARTY (#626)

M/FORM: NAME/ADDRESS THIRD PARTY (#626)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen="ptc(iform,99)
 +s:\$d(nocomp) filen="ptg(903,iform,99) s New=0,rptfile=1
 s U="" i '\$d(DT) s %Gad(SH),NOT="" d NOT s DT=Y k X,Y
 d sac" fctm
 d crinck" fctm

II. WINDOWS

NAME/ADDRESS THIRD PARTY Window I

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
01 NAME	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
M/FORM FID # 6		
28 STORED: Sp("ICC(DA(1),1,0,DA,0),"",1)		
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm "		
5 SETTLEMENT AMOUNT	NUMERIC FORMAT	S:X("S" X=SP(X,"S",2) K:+X'-X(X'7.NI"."2N)!(X>9999999.99)!(X<0) X
M/FORM FID # 7		
28 STORED: Sp("ICC(DA(1),140,DA,0),"",6)		
1 ADDRESS	FREE TEXT	K:SL(X)>70!(SL(X)<1) X
M/FORM FID # 2		
28 STORED: Sp("ICC(DA(1),140,DA,0),"",2)		
44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm=""		
2 CITY	FREE TEXT	K:SL(X)>20!(SL(X)<1) X
M/FORM FID # 3		
28 STORED: Sp("ICC(DA(1),140,DA,0),"",3)		
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm=""		
3 STATE	FOUNDER 1003' DIZ(1003,	
M/FORM FID # 4		
20 CONVERSION CODE: s cm=6s(cm="":":":'Sd("DIZ(1003,cm,0))#10):cm,1:Sp("DIZ(1003,cm,0),"",1))		
28 STORED: Sp("ICC(DA(1),140,DA,0),"",4)		
4 ZIPCODE	FREE TEXT	K:SL(X)>10!(SL(X)<5) X
M/FORM FID # 5		
28 STORED: Sp("ICC(DA(1),140,DA,0),"",5)		
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'Sd(X) cm="" s:cm("" cm=""		

III. POST-FORM CODE

i Sd(file),file s filing=6s('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+1 node0"fmfm

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
515 TOTAL BURIAL EXPENSE	NUMERIC FORMAT	K:+X'-X!(X>9999999)!(X<0)!(X2.EI"."IN.N) X

M/FORM FID # 22

28 STORED: Sp("ICC(DA,505),"",1)

516 BURIAL EXPENSE PAID BY V NUMERIC FORMAT K:+X'-X1(0099999999)!(X<0)!(X2.E1".1N.N) X

M/FORM FID # 23

28 STORED: \$p(ICC(DA,505),"",2)

517 PARTY PAYING FOR BURIAL FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 55

28 STORED: \$p(ICC(DA,506),"",1)

44 POST ID: s %cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

517.01 ADDRESS OF PERSON PAYING FREE TEXT K:SL(X)>45!(SL(X)<1) X

M/FORM FID # 72

28 STORED: \$p(ICC(DA,506),"",2)

44 POST ID: s %cm K:SL(X)>45!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

517.02 CITY OF PER PAYING FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 73

28 STORED: \$p(ICC(DA,506),"",3)

44 POST ID: s %cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

517.03 STATE OF PER PAYING POINTER 1003' DIZ(1003,

M/FORM FID # 74

20 CONVERSION CODE: s cm=\$s(cm="" : "" ,('\$d(DIZ(1003,cm,0))#10):cm,1:\$p(DIZ(1003,cm,0),"",1))

28 STORED: \$p(ICC(DA,506),"",4)

517.04 ZIP CODE OF PER PAYING FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 75

28 STORED: \$p(ICC(DA,506),"",5)

44 POST ID: s %cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

555 ADDRESS OF PERSON SIGNIN FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 77

28 STORED: \$p(ICC(DA,522),"",1)

44 POST ID: s %cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

555.1 CITY OF PERSON SIGNING C FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 78

28 STORED: \$p(ICC(DA,522),"",2)

44 POST ID: s %cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

555.2 STAGE OF PERSON SIGNING POINIER 1003' DIZ(1003,

M/FORM FID # 79

20 CONVERSION CODE: s cm=\$s(cm:"": "", '(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))
28 STORED: \$p('IOC(DA,522),"",3)

555.3 ZIPCODE OF PERSON SIGNIN FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 80

28 STORED: \$p('IOC(DA,522),"",4)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm[""] cm=""

322 DATE EMP/CLAIMANT SIGNED DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 26

20 CONVERSION CODE:
s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm,"000",9,10) " ":" \$e(cm,"000",11,12) s X=X " "
s:X="(1700)" X="" s cm=X

28 STORED: \$p('IOC(DA,303),"",7)

QAS window 5

DATA ELEMENT DATA TYPE POINIER GLOBAL REFERENCE OR INPUT VERIFICATION

520 DOCS NARRATIVE OF REL DI

M/FORM FID # 27

28 STORED: nowhere
29 PREFIELD CODE:
s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shifile=l d filecm="ptrfilel k shifile
i \$d(file),file,New=0 s filing=\$s(' \$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fnfcm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

30 POSTFIELD CODE:
i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: DOC'S NARRATIVE OF INQUIRY (#677)

M/FORM: DOC'S NARRATIVE OF INQUIRY (#677)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen="ptc(iform,99)

```
+s:$d(nocomp) filen="ptg(903,iform,99) s New=0,rptfile=1
s U"" i '$d(DT) s X-$d(SH),NOT="" d "NOT s DT=Y k X,Y
d sec"fufrn
d crinck"fufrn
```

II. WINDOWS

DOC'S NARRATIVE OF INJURY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DOCS NARRATIVE OF REL DI	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p("IOC(DA(1),507,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,l:l:filing)
+d shuffle"fufrn d node0"fufrn
```

521 DOCS TREATMENT/DIAG

M/FORM FID # 28

28 STORED: no-here

29 PREFIELD CODE:

```
s:$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone"ptrfile k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,l:l:filing)
+d upcntr"fufrn s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-CA,DO-CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-CA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)-CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-CA(i)
s:lvl DA-CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: DOCS TREATMENT/DIAGNOSIS (#678)

M/FORM: DOCS TREATMENT/DIAGNOSIS (#678)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen="ptc(iform,99)
+s:$d(nocomp) filen="ptg(903,iform,99) s New=0,rptfile=1
s U"" i '$d(DT) s X-$d(SH),NOT="" d "NOT s DT=Y k X,Y
d sec"fufrn
d crinck"fufrn
```

II. WINDOWS

DOCS TREATMENT/DIAGNOSIS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DOCS TREATMENT/DIAG	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),508,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

522 TREATMENT PRIOR TO DEATH

M/FORM FID # 29

28 STORED: nowhere

29 PREFIELD CODE:

```
s:$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone^ptrfilel k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: TREATMENT PRIOR TO DEATH (#679)

M/FORM: TREATMENT PRIOR TO DEATH (#679)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=^ptc(iform,99)
+s:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s %=$d(SH),%DT="" d ^DT s DT=Y k X,Y
d sec^fmfm
d crinck^fmfm
```

II. WINDOWS

TREATMENT PRIOR TO DEATH Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DES TREATMENT PRIOR TO D	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),509,DA,0),'',1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,l:l:filing)
+d shuffle^fmfm d node0^fmfm
```

523 TREATMENT DATES (PRE-DEA DATE S WDT="E" D ^WDT S X=Y K:Y<1 X

M/FORM FID # 30

20 CONVERSION CODE:

```
s X="(" s:+$e(cm,4,5) X=X $p('JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC','',
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,"",2)'="" X=X " at " $e(cm"000",9,10)' ":" $e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X
```

28 STORED: \$p('ICC(DA,505),'',6)

524 DIRECT CAUSE OF DEATH

M/FORM FID # 31

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(chi(0))>9 s shhfile=l d fileone^ptrfilel k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,l:l:filing)
+d upontr^fmfm s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DC=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DP="" ,DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DP=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: DIRECT CAUSE OF DEATH (#660)

M/FORM: DIRECT CAUSE OF DEATH (#680)

I. PRE-FORM CODE

```
n filen.rptfile,New s:$d(nocomp) filen="ptc(iform,99)
+s:$d(nocomp) filen="ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),WDT="" d ^WDT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm
```

II. WINDOWS

DIRECT CAUSE OF DEATH Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DIRECT CAUSE OF DEATH WORD PROCESSING

M/FORM FID # 1

28 STORED: \$p(^IOC(DA(1),510,DA,0),",",1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

525 CONTRIBUTORY CAUSE OF DE

M/FORM FID # 32

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d fileone^ptrfile1 k shhfile
i \$d(file),file,New=0 s filing-\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f l=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: CONTRIBUTORY CAUSE DEATH (#681)

M/FORM: CONTRIBUTORY CAUSE DEATH (#681)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=^ptc(iform,99)
+s:\$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$H),NOT="" d ^NOT s DT=Y k X,Y
d sec^fmfm
d crlck^fmfm

II. WINDOWS

CONTRIBUTORY CAUSE DEATH Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 CONTRIBUTORY CAUSE OF DE WORD PROCESSING

M/FORM FID # 1

28 STORED: \$p(^IOC(DA(1),511,DA,0),",",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmfm d node0~fmfm
```

526 WAS DEPTH DUE TO WORK	SET OF CODES	Y: YES
		N: NO

M/FORM FID # 33

```
20 CONVERSION CODE: s cm=$p($p("Y: YES;N: NO;","_cm":",2),",",1),cm=$p(cm,"",1)
28 STORED: $p('ICC(DA,505),"",7)
```

526.1 EXPLAIN CAUSAL RELATIONS

M/FORM FID # 34

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone~ptrfilel k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1
```

16 CALLS M/FORM: EXPLAIN CAUSAL RELATIONSHIP (#682)

M/FORM: EXPLAIN CAUSAL RELATIONSHIP (#682)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=~ptc(ifcom,99)
+e:$d(nocomp) filen=~ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '$d(DT) s N=$zd($H),DT="" d ~DT s DT=Y k X,Y
d sec~fmfm
d crinck~fmfm
```

II. WINDOWS

CAUSAL RELATIONSHIP Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN CAUSAL RELATIONS	WORD PROCESSING	

M/FORM FID # 1

```
28 STORED: $p('ICC(DA(1),512,DA,0),"",1)
```

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmfm d nodes0~fmfm

527 AUTOPSY, BIOPSY PERFORMED SET OF CODES Y: YES
N: NO

M/FORM FID # 35

20 CONVERSION CODE: s cm=\$p(\$p("Y: YES;N: NO;"_"_cm_"",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^IOC(DA,505),"",8)

401 REFERRAL PHYSICIAN FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 61

28 STORED: \$p(^IOC(DA,401),"",1)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm("" cm=""

401.1 RP ADDRESS FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 62

28 STORED: \$p(^IOC(DA,401),"",2)
44 POST ID: s X=cm K:\$L(X)>45!(\$L(X)<1) X s:\$d(X) cm="" s:cm("" cm=""

401.2 RP CITY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 63

28 STORED: \$p(^IOC(DA,401),"",3)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm("" cm=""

401.3 RP STATE POINTER 1003' DIZ(1003,

M/FORM FID # 64

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^IOC(DA,401),"",4)

401.4 RP ZIPCODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 65

28 STORED: \$p(^IOC(DA,401),"",5)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:\$d(X) cm="" s:cm("" cm=""

400 ATTENDING PHYSICIAN FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 66

28 STORED: \$p(^IOC(DA,400),"",1)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm("" cm=""

400.9 DATE SIGNED

DATE

S ^DT="E" D ^DT S X-Y K:Y<1 X

M/FORM FID # 71

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=K \$e(cm,6,7) " " s X=K (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=K " at " \$e(cm^"000",9,10) _:" \$e(cm^"000",11,12) s X=K ")"
s:X="(1700)" X=" s cm=K

28 STORED: \$p(^IOC(DA,400),"",10)

400.1 DOCS ADDRESS

FREE TEXT

K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 67

28 STORED: \$p(^IOC(DA,400),"",2)

44 POST ID: s X=cm K:\$L(X)>45!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

400.2 DOCS CITY

FREE TEXT

K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 68

28 STORED: \$p(^IOC(DA,400),"",3)

4' POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

400.3 DOCS STATE

POINTER 1003' DIZ(1003,

M/FORM FID # 69

20 CONVERSION CODE: s cm=\$e(cm="":",",(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^IOC(DA,400),"",4)

400.4 DOCS ZIPCODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 70

28 STORED: \$p(^IOC(DA,400),"",5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file,new=0 s:\$d(lvl) lvl=0
+s filing=\$e(\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^filing
i file=1!(\$d(ENFLAG)) d ^T3FILE

CA-5B CLAIM FOR COMPENSATION BY RELATIVES

1. NAME OF DESEASED _____
2. DATE OF BIRTH _____ 3. DATE OF INJURY _____
4. DATE OF DEATH _____ 5. SOCIAL SECURITY NUMBER _____
6. NAME OF EMPLOYING AGENCY _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
7. NATURE OF INJURY WHICH CAUSED DEATH _____
- *Press <Return> for NATURE OF INJURY/ILLNESS*
8. NAME OF DEPENDENT _____
9. DEPENDENT'S ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
10. DEPENDENT'S BIRTHDATE _____
11. DEPENDENT'S OCCUPATION _____
12. DEPENDENT'S SOCIAL SECURITY NUMBER _____
13. DEPENDENT'S RELATIONSHIP TO EMPLOYEE _____
14. EXTENT OF DEPENDENCY ON EMPLOYEE _____
15. TOTAL AMOUNT EMPLOYEE CONTRIBUTED TO DEPENDENT'S SUPPORT DURING 12 MONTHS IMMEDIATELY PRIOR TO DEATH _____
16. DID DEP LIVE WITH EMP 12 MOS PRIOR DEATH _____

CLAIM FOR COMPENSATION BY RELATIVES (PAGE 2)

17. TOTAL AMOUNT EMPLOYEE PAID DEPENDENT IN MONEY OR SERVICE FOR ROOM AND BOARD IN ADDITION TO AMOUNT SHOWN IN #15.
PER _____
18. IF NO FIXED AMOUNT WAS PAID FOR ROOM AND BOARD, WHAT IS THE FAIR VALUE OF SUCH ROOM AND BOARD? PER _____
19. IF DEPENDENT WAS EMPLOYED DURING 12 MONTHS PRIOR TO EMPLOYEE DEATH GIVE:
TYPE OF WORK PERFORMED: _____
PERIOD OF EMPLOYMENT: _____
MONTHLY PAY RATE: _____
NAME OF EMPLOYER _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
20. DEPENDENT'S INCOME: INVESTMENTS _____ PENSIONS _____
PERSONS OTHER THAN EMPLOYEE _____
OTHER _____ TOTAL _____

INFORMATION ABOUT DEPENDENT'S HUSBAND OR WIFE (ITEM 21 THROUGH 25)

21. BIRTHDATE _____ 22. OCCUPATION _____
23. MONTHLY PAY RATE _____
24. TOTAL INCOME DURING 12 MOS PRIOR DEATH _____
25. LIST PROPERTY OWNED BY DEPENDENT _____
- Press <Return> for DEPENDENT'S PROPERTY _____

CA-5B CLAIM FOR COMPENSATION BY RELATIVES (PAGE 3)

26. IF EMPLOYEE WAS EVER IN THE ARMED FORCES OF THE UNITED STATES, GIVE:
SERVICE NUMBER: _____
BRANCH OF SERVICE: _____
PERIOD OF SERVICE: _____
27. VA CLAIM NUMBER: _____
VA OFFICE WHERE CLAIM FILED _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
28. CPFS CLAIM NUMBER: _____ DATE ANNUITY BEGAN: _____
AMOUNT PAID PER MONTH: \$ _____
29. IF THIRD PARTY CLAIM, Press <Return> for NAME/ADDRESS THIRD PARTY _____
30. TOTAL BURIAL EXPENSE _____ 31. BURIAL EXPENSE PAYABLE BY VA \$ _____
32. PARTY PAYING FOR BURIAL (OTHER THAN VA) _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
AMOUNT PAID: _____
33. SIGNATURE OF PERSON FILING CLAIM _____
34. ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
35. DATE _____

CLAIM FOR COMPENSATION BY RELATIVES (PAGE 4)

ATTENDING PHYSICIAN'S REPORT

1. NAME OF DESEASED EMPLOYEE (SEE #1 PAGE 1) _____
2. DATE OF DEATH (SEE #4 PAGE 1) _____
3. WHAT HISTORY OF INJURY RELATED DISEASE?
Press <Return> for DOC'S NARRATIVE OF INJURY _____
4. IF TREATED FOR DISEASE, GIVE DIAGNOSIS.
Press <Return> for DOC'S TREATMENT/DIAGNOSIS _____
5. IF DEATH WAS NOT INSTANTANEOUS, DESCRIBE
Press <Return> for TREATMENT PRIOR TO DEATH _____
6. SHOW DATES ON WHICH TREATMENT WAS GIVEN _____
7. *Press <Return> for DIRECT CAUSE OF DEATH* _____
8. *Press <Return> for CONTRIBUTORY CAUSE OF DEATH* _____
9. WAS DEATH DUE TO THE INJURY REPORT #3? _____
MEDICAL REASON FOR OPINION _____
Press <Return> for EXPLAIN CAUSAL RELATIONSHIP _____
10. WAS A BIOPSY OR AUTOPSY PERFORMED? _____
11. ATTENDING PHYSICIAN _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
13. DATE SIGNED _____

_____ CA-5B CON'T _____

_____ DEPENDENT'S PROPERTY _____
_____ (OMIT CLOTHING, FURNITURE, PERSONAL ITEMS) _____

DEPENDS PROPERTY _____
DATE ACQUIRED _____ VALUE _____

_____ THIRD PARTY INFORMATION _____

NAME _____ SETTLEMENT AMOUNT _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

*Word Processing Window will appear.

M/FORM: OA-5B (#688)

I. PRE-FORM CODE

```
n file,nptfile,filing s:'$d(nocomp) file=ptc(ifom,99)
+s:$d(nocomp) file=ptg(903,ifom,99) s New=0,nptfile=1,ideast=2 k ident
s U="" i '$d(DT) s X=$d(SH),%DT="" d %DT s DT=Y k X,Y
d sec`mfom
```

II. WINDOWS

OAS Window I

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	POINTER 1004'	EMPLOY(
--------	---------------	---------

M/FORM FID # 104

```
17 EXTERNAL DEFAULT: NAME
18 INTERNAL DEFAULT: NAMEPTR
20 CONVERSION CODE: s cm=$e(cm:"",',($d(EMPLOY(cm,0))#10):cm,1:Sp(EMPLOY(cm,0),"",1))
28 STORED: Sp(IOC(DA,0),"",3)
32 POST FILING CODE: d setirck`mfom
44 POST ID: I $d(new) s:'$d(IOC("C",new,x1)) cm="",y=24,msg="CLAIM NOT LOGGED!"
54 INDEX: IOC("C")
55 LOOKUP CODE:
104~20~Sp(IOC(Lids,0),"",3)
3~23~Sp(IOC(Lids,0),"",2)
```

6 BIRTHDATE	DATE	S %DT="E" D %DT S %Y K:Y<1 X
-------------	------	------------------------------

M/FORM FID # 2

```
20 CONVERSION CODE:
s X=("$ s:+$e(cm,4,5) X=X Sp(JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) ", " s X=X (1700+$e(cm,1,3))
s:Sp(cm,"",2)'="" X=X " at " $e(cm"000",9,10) ":" $e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X
28 STORED: Sp(IOC(DA,0),"",7)
```

1 INJURY/ILLNESS DATE & TI	DATE	S %DT="E" D %DT S %Y K:Y<1 X I \$d(X) K:\$P(X,"",1)>DT X
----------------------------	------	--

M/FORM FID # 3

```
20 CONVERSION CODE:
s X=("$ s:+$e(cm,4,5) X=X Sp(JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) ", " s X=X (1700+$e(cm,1,3))
s:Sp(cm,"",2)'="" X=X " at " $e(cm"000",9,10) ":" $e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X
28 STORED: Sp(IOC(DA,0),"",2)
32 POST FILING CODE: d setirck`mfom
```

54 INDEX: ^IOC(^D)
 55 LOOKUP NODE:
 3^23^Sp(^IOC(Lids,0),^",2)
 104^20^Sp(^IOC(Lids,0),^",3)

103 DATE OF DEATH DATE S ^DT="ET" D ^NOT S X=Y K:Y<1 X I \$D(X) K:X<\$P(^IOC(DA,0),U,2) X

M/FORM FID # 61

20 CONVERSION CODE:
 s X="(" s:+Se(cm,4,5) X=X Sp(^JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",^",
 +Se(cm,4,5)) ^" s:+Se(cm,6,7) X=X Se(cm,6,7) ^", s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)^" X=X " at ^Se(cm^"000",9,10)^":^Se(cm^"000",11,12) s X=X ^"
 s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,100),^",4)

3 SSN FREE TEXT K:SL(X)>9!(SL(X)<9)!^(X^9N) X

M/FORM FID # 5

28 STORED: Sp(^IOC(DA,0),^",4)
 44 POST ID: s X=cm K:SL(X)>9!(SL(X)<9)!^(X^9N) X s:\$d(X) cm="" s:cm(^"" cm=""

13 DUTY STATION POINTER 1062' INJ(1062,

M/FORM FID # 62

20 CONVERSION CODE: s cm=Se(cm^":^",^(\$d(^INJ(1062,cm,0))#10):cm,1:Sp(^INJ(1062,cm,0),^",1))
 28 STORED: Sp(^IOC(DA,9),^",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:SL(X)>5..(SL(X)<1) X

M/FORM FID # 63

28 STORED: Sp(^IOC(DA,9),^",5)
 44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:\$d(X) cm="" s:cm(^"" cm=""

13.5 DUTY STATION CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 64

28 STORED: Sp(^IOC(DA,9),^",6)
 44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:\$d(X) cm="" s:cm(^"" cm=""

13.6 DUTY STATION STATE POINTER 1003' DIEZ(1003,

M/FORM FID # 65

20 CONVERSION CODE: s cm=Se(cm^":^",^(\$d(^DIEZ(1003,cm,0))#10):cm,1:Sp(^DIEZ(1003,cm,0),^",1))
 28 STORED: Sp(^IOC(DA,9),^",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 66

28 STORED: \$p(^ICC(DA,9), "", 2)

44 POST ID: s X=cm K:SL(X)>101(SL(X)<5) X s:\$d(X) cm="" s:cm["" cm=""

101.1 NATURE OF INJURY EXPLANA

M/FORM FID # 7

28 STORED: nowhere

29 PREFIELD CODE:

```
s:$d(lvl) lvl=0 i $d(cbi(0))>9 s sthfile=1 d filecne^ptrfile1 k sthfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fncm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DP="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: NATURE OF INJURY/ILLNESS (#642)

M/FORM: NATURE OF INJURY/ILLNESS (#642)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nccomp) filen=ptc(iform,99)
+s:$d(nccomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),WDT="" d "WDT s DT=y k X,Y
d sec^fncm
d crinch^fncm
```

II. WINDOWS

NATURE OF INJURY/ILLNESS Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NATURE OF INJURY EXPLANA	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),101,DA,0), "", 1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fncm d nodes0^fncm
```

503 OTHER DEPEND AT DEATH	FREE TEXT	K:SL(X)>301(SL(X)<1) X
---------------------------	-----------	------------------------

M/FORM FID # 8

28 STORED: \$p(^ICC(DA,502),"",5)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

504 OTHER DEPEND ADDRESS AT FREE TEXT K:\$L(X)>55!(\$L(X)<1) X

M/FORM FID # 47

28 STORED: \$p(^ICC(DA,502),"",1)

44 POST ID: s X=cm K:\$L(X)>55!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

504.1 OTHER DEPEND CITY FREE TEXT K:\$L(X)>23!(\$L(X)<1) X

M/FORM FID # 48

28 STORED: \$p(^ICC(DA,502),"",2)

44 POST ID: s X=cm K:\$L(X)>23!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

504.2 OTHER DEPEND STATE POINTER 1003' DIZ(1003,

M/FORM FID # 57

20 CONVERSION CODE: s cm=\$s(cm="":'(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

22 STORED: \$p(^ICC(DA,502),"",3)

504.3 OTHER DEPEND ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 50

28 STORED: \$p(^ICC(DA,502),"",4)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

539 OTHER DEPEND DOB DATE S WDT="E" D ^WDT S X=Y K:Y<1 X

M/FORM FID # 10

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at "\$e(cm,"000",9,10)'="\$e(cm,"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,515),"",5)

540 OTHER DEPEND OCCUPATION POINTER 1001' DIZ(1001,

M/FORM FID # 58

20 CONVERSION CODE: s cm=\$s(cm="":'(\$d(^DIZ(1001,cm,0))#10):cm,1:\$p(^DIZ(1001,cm,0),"",2))

28 STORED: \$p(^ICC(DA,515),"",6)

541 OTHER DEPEND SSN FREE TEXT K:\$L(X)>9!(\$L(X)<9)!'(X?9N) X

M/FORM FID # 59

28 STORED: \$p('ICC(DA,515),'",7)

44 POST ID: s X=cm K:\$L(X)>9!(\$L(X)<9)!'(X?9N) X s:'\$d(X) cm="" s:cm["" cm=""

542 OTHER DEPEND RELATIONSHI FREE TEXT K:\$L(X)>30!(\$L(X)<3) X

M/FORM FID # 13

28 STORED: \$p('ICC(DA,515),'",8)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<3) X s:'\$d(X) cm="" s:cm["" cm=""

543 PARTIAL/TOTAL DEPENDENCY SET OF CODES T:TOTAL
P:PARTIAL

M/FORM FID # 52

20 CONVERSION CODE: s cm=\$p(\$p("T:TOTAL;P:PARTIAL;";" _cm ":"2);";1),cm=\$p(cm,"",1)

28 STORED: \$p('ICC(DA,515),'",9)

544 AMT CONTRIB DEP PRIOR TO NUMERIC FORMAT S:X["\$" X=\$p(X,"\$",2) K:+X'=>X(X'?.N1"."2N)!(X>9999999.99)!(X<.01) X

M/FORM FID # 15

28 STORED: \$p('ICC(DA,515),'",10)

545 DEP W/EMP 12 MOS PRIOR T SET OF CODES Y:YES
N:NO

M/FORM FID # 16

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;";" _cm ":"2);";1),cm=\$p(cm,"",1)

28 STORED: \$p('ICC(DA,520),'",1)

CASE Window 2

DATA ELEMENT	DATA TYPE	FOINIER GLOBAL REFERENCE OR INPUT VERIFICATION
546 TOTAL ROOM/BOARD BASIS	NUMERIC FORMAT	S:X["\$" X=\$p(X,"\$",2) K:+X'=>X(X'?.N1"."2N)!(X>999999)!(X<0) X

M/FORM FID # 17

28 STORED: \$p('ICC(DA,520),'",2)

29 PREFIELD CODE:

i '\$d(cbl(16)) s skip="D"

s:cbl(16)["n"!cbl(16)["N"] skip="D"

546.1 ROOM/BOARD BASIS SET OF CODES W:WEEKLY
M:MONTHLY
B:BI-WEEKLY
A:ANNUALLY
D:DAILY


```

20 CONVERSION CODE:  s am=$p($p";W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;A:ANNUALLY;
                    D:DAILY;";" am ":"2);";1),am=$p(am,";",1)
28 STORED:  $p(^IOC(DA,520),"",5)
29 PREFIELD CODE:
    i '$d(cbi(17)) s skip="D"
    s:cbi(16)["n"]:(cbi(16)["N"]) skip="D"

```

M/FORM FID # 14

547.1 VALLE BASIS

M/FORM FID # 54

M/FORM FID # 67

M/PCN: FID # 68

M/FORM FID # 69

A-108

548.3 NAME OF EMPLOYER FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 70

28 STORED: \$p(^IOC(DA,523),"",4)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""

548.4 ADDRESS OF EMPLOYER FREE TEXT K:\$L(X)>60!(\$L(X)<1) X

M/FORM FID # 71

28 STORED: \$p(^IOC(DA,523),"",5)

44 POST ID: s X=cm K:\$L(X)>60!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""

548.5 CITY OF EMPLOYER FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 72

28 STORED: \$p(^IOC(DA,523),"",6)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""

548.6 STATE OF EMPLOYER FOURIER 1003' DIZ(1003,

M/FORM FID # 73

20 CONVERSION CODE: s cm=\$s(cm:"":,"",(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^IOC(DA,523),"",7)

548.7 ZIP CODE OF EMPLOYER FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 74

28 STORED: \$p(^IOC(DA,523),"",8)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:\$d(X) cm="" s:cm[""] cm=""

549 INVESTMENTS NUMERIC FORMAT S:X["\$"] X=\$P(X,"\$",2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<0) }

M/FORM FID # 75

28 STORED: \$p(^IOC(DA,524),"",1)

549.1 PENSIONS OF OTHER DEPEND NUMERIC FORMAT S:X["\$"] X=\$P(X,"\$",2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<0) }

M/FORM FID # 76

28 STORED: \$p(^IOC(DA,524),"",2)

549.2 PERSONS OTHER THAN EMPLO NUMERIC FORMAT S:X["\$"] X=\$P(X,"\$",2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<0) }

M/FORM FID # 77

28 STORED: \$p(^IOC(DA,524),"",3)

549.3 OTHER INCOME FOR DEPEND NUMERIC FORMAT S:X["\$"] X=\$P(X,"\$",2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<0) }

M/FORM FID # 78

28 STORED: \$p(^ICC(DA,524),"",4)

549.4 TOTAL DEPENDENT INCOME NUMERIC FORMAT S:X("\$" X=\$P(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999999.99)!(X<0) X

M/FORM FID # 79

28 STORED: \$p(^ICC(DA,524),"",5)

550 DEPEND SPOUSE DOB DATE S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 21

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm"000",9,10) " " \$e(cm"000",11,12) s X=X "_"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,520),"",6)

551 DEPEND SPOUSE OCCUPATION POINTER 1001' DIZ(1001,

M/FORM FID # 60

20 CONVERSION CODE: s cm=\$s(cm:"":",",'(\$d(^DIZ(1001,cm,0))#10):cm,1:\$p(^DIZ(1001,cm,0),"",2))
28 STORED: \$p(^ICC(DA,520),"",7)

552 DEPEND SPOUSE MONTHLY WA FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 23

28 STORED: \$p(^ICC(DA,520),"",8)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm(" " cm="

553 DEPEND INCOME 12 MOS PRI NUMERIC FORMAT S:X("\$" X=\$P(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>99999999.99)!(X<.01) X

M/FORM FID # 24

28 STORED: \$p(^ICC(DA,520),"",9)

554 DEPENDS PROPERTY

M/FORM FID # 25

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s sthfile=1 d filecm^ptrfile1 k sthfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)=" k file
s lvl=lvl+1

```
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: DEPENDENT'S PROPERTY (#720)

M/FORM: DEPENDENT'S PROPERTY (#720)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),DT="" d ~DT s DI=Y k X,Y
d sec`fmfm
```

II. WINDOWS

DEPENDENT'S PROPERTY WINDOW 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DEPENDS PROPERTY	FREE TEXT	K:SL(X)>40!(SL(X)<1) X
----------------------	-----------	------------------------

M/FORM FID # 1

```
28 STORED: Sp('TOC(DA(1),521,DA,0),"",1)
32 POST FILING CODE: d aetirx`fmfm
44 POST ID: s X=on K:SL(X)>40!(SL(X)<1) X s:$d(X) on="" s:cm[""" on=""
54 INDEX: 'TOC(DA(1),521,"B")
55 LOOKUP CODE: 1'40`Sp('TOC(DA(1),521,lids,0),"",1)
```

1 DATE ACQUIRED	FREE TEXT	K:SL(X)>10!(SL(X)<1) X
-----------------	-----------	------------------------

M/FORM FID # 2

```
28 STORED: Sp('TOC(DA(1),521,DA,0),"",2)
44 POST ID: s X=on K:SL(X)>10!(SL(X)<1) X s:$d(X) on="" s:cm[""" on=""
```

2 VALUE	NUMERIC FORMAT	S:X"\$" X=\$P(X,"\$",2) K:+X'~X(X'7.11".2N)!(X>9999999.99)!(X<.01) X
---------	----------------	--

M/FORM FID # 3

```
28 STORED: Sp('TOC(DA(1),521,DA,0),"",3)
```

III. POST-FORM CODE

```
i $d(file),file s filing-$d('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d nodes0`fmfm
```

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
505 SERVICE NO	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 26

28 STORED: \$p(^ICC(DA,500),"",3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

506 BRANCH OF SERVICE	SET OF CODES	A:ARMY N:NAVY M:MARINES F:AIR FORCE C:COAST GUARD
-----------------------	--------------	---

M/FORM FID # 27

28 STORED: \$p(^ICC(DA,500),"",4)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

507 PERIOD OF SERVICE	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
-----------------------	-----------	--------------------------

M/FORM FID # 28

28 STORED: \$p(^ICC(DA,500),"",5)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710 VA CLAIM NUMBER	FREE TEXT	K:\$L(X)>15!(\$L(X)<1) X
---------------------	-----------	--------------------------

M/FORM FID # 85

28 STORED: \$p(^ICC(DA,701),"",1)

44 POST ID: s X=cm K:\$L(X)>15!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710.1 VA OFFICE WHERE CLAIM FI	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
--------------------------------	-----------	--------------------------

M/FORM FID # 86

28 STORED: \$p(^ICC(DA,701),"",2)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710.2 VA STREET ADDRESS	FREE TEXT	K:\$L(X)>54!(\$L(X)<1) X
-------------------------	-----------	--------------------------

M/FORM FID # 87

28 STORED: \$p(^ICC(DA,701),"",3)

44 POST ID: s X=cm K:\$L(X)>54!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710.3 VA CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
---------------	-----------	--------------------------

M/FORM FID # 88

28 STORED: \$p('ICC(DA,701),'',4)
 44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm('' cm=""

710.4 VA STATE

POINTER 1003' DIEZ(1003,

M/FORM FID # 89

20 CONVERSION CODE: s cm=\$s(cm="":'',('\$d('DIEZ(1003,cm,0))#10):cm,1:\$p('DIEZ(1003,cm,0),'',1))
 28 STORED: \$p('ICC(DA,701),'',5)

710.5 VA ZIP CODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 90

28 STORED: \$p('ICC(DA,701),'',6)
 44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm('' cm=""

714 OTHER ANNUITIES CLAIM NU

FREE TEXT

K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 91

28 STORED: \$p('ICC(DA,707),'',1)
 44 POST ID: s X=cm K:\$L(X)>15!(\$L(X)<1) X s:'\$d(X) cm="" s:cm('' cm=""

714.1 OTHER ANNUITY DATE BEGN

DATE

S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 92

20 CONVERSION CODE:
 s X="(" s:+\$s(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +\$s(cm,4,5)) " " s:+\$s(cm,6,7) X=X \$s(cm,6,7) ", " s X=X (1700+\$s(cm,1,3))
 s:\$p(cm,".",2)'="" X=X " at " \$s(cm"000",9,10) ":" \$s(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: \$p('ICC(DA,707),'',2)

714.2 OTHER ANNUITY AMOUNT

NUMERIC FORMAT

S:X("\$" X=\$p(X,"\$",2) K:+X'-%X(X'?.N1"."2N)!(X09999999.99)!(X<0))

M/FORM FID # 93

28 STORED: \$p('ICC(DA,707),'',3)

141.1 THIRD PARTY

M/FORM FID # 31

28 STORED: no where

29 PREFIELD CODE-

s:'\$d(lvl) ""=0 i \$d(cbi(0))>9 s shfile=1 d filecm=ptrfilel k shfile
 i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
 +d upcntr=factm s New=0,cbi(0)="" k file
 s lvl=lvl+1
 _ lvl s DA(0)=CA,CO=CA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=CA(i-1)
 ? DA(0) S DA="" ,DI=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: NAME/ADDRESS THIRD PARTY (#626)

M/FORM: NAME/ADDRESS THIRD PARTY (#626)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),DT="" d ~DT s DT=Y k X,Y
d sec~fmfm
d crindx~fmfm

II. WINDOWS

NAME/ADDRESS THIRD PARTY Window I

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 6		
28 STORED: \$p('ICC(DA(1),140,DA,0),"",1)		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""		
5 SETTLEMENT AMOUNT	NUMERIC FORMAT	S:X["\$" X=\$P(X,"\$",2) K:X'~X(X'?..N1". "2N)!(X>99999999.99)!(X<0) >
M/FORM FID # 7		
28 STORED: \$p('ICC(DA(1),140,DA,0),"",6)		
1 ADDRESS	FREE TEXT	K:\$L(X)>70!(\$L(X)<1) X
M/FORM FID # 2		
28 STORED: \$p('ICC(DA(1),140,DA,0),"",2)		
44 POST ID: s X=cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""		
2 CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
M/FORM FID # 3		
28 STORED: \$p('ICC(DA(1),140,DA,0),"",3)		
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""		
3 STATE	POINTER 1003' DIZ(1003,	
M/FORM FID # 4		
20 CONVERSION CODE: s cm=\$s(cm="":","('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))		

28 STORED: \$p(^TCC(DA(1),140,DA,0),"",4)

4 ZIPCODE

FREE TEXT

K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 5

28 STORED: \$p(^TCC(DA(1),140,DA,0),"",5)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

III. POST-FORM CODE

1 \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d mode0~finfra

515 TOTAL BURIAL EXPENSE

NUMERIC FORMAT

K:+X'-KI(X09999999)!(X<0)!(X?.E1"."IN,N) X

M/FORM FID # 32

28 STORED: \$p(^TCC(DA,505),"",1)

516 BURIAL EXPENSE PAID BY V

NUMERIC FORMAT

K:+X'-KI(X09999999)!(X<0)!(X?.E1"."IN,N) X

M/FORM FID # 33

28 STORED: \$p(^TCC(DA,505),"",2)

517 PARTY PAYING FOR BURIAL

FREE TEXT

K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 80

28 STORED: \$p(^TCC(DA,506),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

517.01ADDRESS OF PERSON PAYING

FREE TEXT

K:SL(X)>45!(SL(X)<1) X

M/FORM FID # 81

28 STORED: \$p(^TCC(DA,506),"",2)

44 POST ID: s X=cm K:SL(X)>45!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

517.02CITY OF PER PAYING

FREE TEXT

K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 82

28 STORED: \$p(^TCC(DA,506),"",3)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

517.03STATE OF PER PAYING

FOUNTER 1003' DIZ(1003,

M/FORM FID # 83

20 CONVERSION CODE: s cm-\$s(cm:"":",('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))
28 STORED: \$p(^TCC(DA,506),"",4)

517.04 ZIP CODE OF PER PAYING FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 84

28 STORED: Sp(^IOC(DA,506),"",5)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm(" " cm=""

517.1 AMOUNT OTHER PARTY PAID NUMERIC FORMAT S:X["\$"] X=Sp(X,"\$",2) K:+X'=X(X'?.NL"."2N)!(X>9999999.99)!(X<.01) X

M/FORM FID # 35

28 STORED: Sp(^IOC(DA,505),"",3)

555 ADDRESS OF PERSON SIGNIN FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 94

28 STORED: Sp(^IOC(DA,522),"",1)

44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

555.1 CITY OF PERSON SIGNING C FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 95

28 STORED: Sp(^IOC(DA,522),"",2)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

555.2 STATE OF PERSON SIGNING POINTER 1003' DIZ(1003,

M/FORM FID # 96

20 CONVERSION CODE: s cm=Se(cm="":",',(\$d(DIZ(1003,cm,0))#10):cm,1:Sp(DIZ(1003,cm,0),"",1))

28 STORED: Sp(^IOC(DA,522),"",3)

555.3 ZIP CODE OF PERSON SIGNIN FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 97

28 STORED: Sp(^IOC(DA,522),"",4)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm(" " cm=""

322 DATE EMP/CLAIMANT SIGNED DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 37

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC"," ",+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3)) s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10)" : " Se(cm"000",11,12) s X=X " s:X="(1700)" X="" s cm=X

28 STORED: Sp(^IOC(DA,303),"",7)

CASB Window 4

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

520 DOCS NARRATIVE OF REL DI

M/FORM FID # 38

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone`ptrfilel k shhfile
i $d(file),file,New'=0 s filing=$e('$d(filing):lvl+1,lvl'<filing:lvl+1,l:1:filing)
+d upcntr`fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: DOC'S NARRATIVE OF INQUIRY (#677)

M/FORM: DOC'S NARRATIVE OF INQUIRY (#677)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(mocmp) filen=`ptc(iform,99)
+s:$d(mocmp) filen=`ptg(903,iform,99) : New=0,rptfile=1
s Ua"" i '$d(DT) s %=$d(SH),%DT="" d `&DT s DT=Y k X,Y
d sec`fufm
d crinck`fufm
```

II. WINDOWS

DOC'S NARRATIVE OF INQUIRY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DOCS NARRATIVE OF REL DI WORD PROCESSING

M/FORM FID # 1

28 STORED: Sp('ICC(DA(1),507,DA(0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$e('$d(filing):lvl,lvl'<filing:lvl,l:1:filing)
+d shuffle`fufm d node0`fufm
```

521 DOCS TREATMENT/DIAG

M/FORM FID # 39

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone'ptrfilel k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr'fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA:lvl,DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: DOCS TREATMENT/DIAGNOSIS (#678)

M/FORM: DOCS TREATMENT/DIAGNOSIS (#678)

I. PRE-FORM CODE

```
n filen,'ptfile,New s:'$d(nocomp) filen='ptc(ifom,99)
+s:$d(nocomp) filen='ptg(903,ifom,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),NOT="" d "NOT s DT=Y k X,Y
d sec'fufm
d crindx'fufm
```

II. WINDOWS

DOCS TREATMENT/DIAGNOSIS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DOCS TREATMENT/DIAG	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p('DOC(DV(1),508,DA,0),"",1)		

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl+1,1:filing)
+d shuffle'fufm d node0'fufm
```

522 TREATMENT PRIOR TO DEMON

M/FORM FID # 40

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone'ptrfilel k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr'fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

k DA(0) S DA="",DI=""

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: TREATMENT PRIOR TO DEATH (#679)

M/FORM: TREATMENT PRIOR TO DEATH (#679)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=ptc(ifcom,99)
+s:\$d(nocomp) filen=ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$H),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm
d crink`fmfm

II. WINDOWS

TREATMENT PRIOR TO DEATH WINDOW I

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DES TREATMENT PRIOR TO D	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(ICC(DA(1),509,DA(0),"",1))

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

523 TREATMENT DATES (PRE-DEA	DATE	S %DT="E" D %DT S X=Y R:Y<1 X
------------------------------	------	-------------------------------

M/FORM FID # 41

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm"000",9,10) _:" \$e(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(ICC(DA,505),"",6)

524 DIRECT CAUSE OF DEATH

M/FORM FID # 42

28 STORED: nowhere

29 PREFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone~ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

```

16 CALLS M/FORM: DIRECT CAUSE OF DEATH (#680)

M/FORM: DIRECT CAUSE OF DEATH (#680)

I. PRE-FORM CODE

```

n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),YDT="" d ~NOT s DT=Y k X,Y
d sec~fmfm
d crink~fmfm

```

II. WINDOW

DIRECT CAUSE OF DEATH Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DIRECT CAUSE OF DEATH	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p~ICC(DA(1),510,DA(0),"",1)		

III. POST-FORM CODE

```

i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle~fmfm d node0~fmfm

```

525 CONTRIBUTORY CAUSE OF DE

M/FORM FID # 43

28 STORED: nowhere

29 PREFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone~ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""

```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: CONTRIBUTORY CAUSE DEATH (#681)

M/FORM: CONTRIBUTORY CAUSE DEATH (#681)

I. PRE-FORM CODE

```
n file, rptfile, New s:$d(nocomp) file=~ptc(ifom,99)
+s:$d(nocomp) file=~ptg(903,ifom,99) s New=0, rptfile=1
s U="" i '$d(DT) s %$ad($H),%DT="" d ~MT s DT=Y k X,Y
d sec~fufm
d crinck~fufm
```

II. WINDOWS

CONTRIBUTORY CAUSE DEATH Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 CONTRIBUTORY CAUSE OF DE	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(~TCC(DA(1),511,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fufm d node0~fufm
```

526 WAS DEATH DUE TO WORK	SET OF CODES	Y: YES N: NO
---------------------------	--------------	-----------------

M/FORM FID # 44

20 CONVERSION CODE: s cm=\$p(\$p("Y: YES;N: NO;",";","_cm_":",2)",";"),1),cm=\$p(cm,";",1)
28 STORED: \$p(~TCC(DA,505),"",7)

526.1 EXPLAIN CAUSAL RELATIONS

M/FORM FID # 45

28 STORED: nowhere

29 PREFIELD CODE:

```
s:$d(lvl) lvl=0 i $d(chi(0))>9 s shifile=1 d filecm~ptrfile k shifile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr~fufm s New=0,chi(0)= "" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

i lvl s DA(0)=DA t i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: EXPLAIN CAUSAL RELATIONSHIP (#682)

M/FORM: EXPLAIN CAUSAL RELATIONSHIP (#682)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=^ptc(iform,99)
+s:\$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$ad(SH),NOT="" d ^NOT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm

II. WINDOWS

CAUSAL RELATIONSHIP Window I

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 EXPLAIN CAUSAL RELATIONS	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),512,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

527 AUTOPSY/BIOPSY PERFORMED	SET OF CODES	Y:YES N:NO
------------------------------	--------------	---------------

M/FORM FID # 55

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"", "_cm_":",2"),",",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,505),"",8)

400 ATTENDING PHYSICIAN	FREE TEXT	K:SL(X)>301:(SL(X)<1) X
-------------------------	-----------	-------------------------

M/FORM FID # 98

28 STORED: \$p(^ICC(DA,400),"",1)
44 POST ID: s X=cm K:SL(X)>301:(SL(X)<1) X s:\$d(X) cm="" s:cm[""" cm=""

400.1 DOCS ADDRESS	FREE TEXT	K:SL(X)>54:(SL(X)<1) X
--------------------	-----------	------------------------

M/FORM FID # 99

28 STORED: \$p(^ICC(DA,400),"",2)

44 POST ID: s X=cm K:\$L(X)>45!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

400.2 DOCS CITY

FREE TEXT

K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 100

28 STORED: \$p(^ICC(DA,400), "", 3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

400.3 DOCS STATE

FOUNDER 1003' DIZ(1003,

M/FORM FID # 101

20 CONVERSION CODE: s cm=\$e(cm="" : "" , '(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0), "", 1))

28 STORED: \$p(^ICC(DA,400), "", 4)

400.4 DOCS ZIPCODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 102

28 STORED: \$p(^ICC(DA,400), "", 5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

400.9 DATE SIGNED

DATE

S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 103

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ".
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " , " s X=X (1700+\$e(cm,1,3))
s:\$p(cm, ".", 2)'="" X=X " at " \$e(cm "000", 9,10) _:" \$e(cm "000", 11,12) s X=X _)"
s:X="(1700)" :="" s cm=X

28 STORED: \$p(^ICC(DA,400), "", 10)

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fmfm
i file=1!(\$d(ERFLAG)) d ^T3FILE

CA-6 OFFICIAL SUPERIOR'S REPORT OF EMPLOYEE'S DEATH

1. NAME OF DESEASED _____
2. DATE OF BIRTH _____ 3. SEX _____
4. SOCIAL SECURITY NUMBER _____
5. DEPARTMENT OR AGENCY _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
6. OWCP AGENCY CODE _____ 7. OSHI SITE CODE _____
8. REPORTING OFFICE _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
9. NAME OF EMPLOYEE'S OFFICIAL SUPERVISOR _____
10. DATE AND HOUR OF INJURY _____
11. DATE AND HOUR OF DEATH _____
12. DATE AND HOUR EMPLOYEE'S PAY STOPPED _____
13. DESCRIBE HOW INJURY OCCURRED *Press <Return> for CAUSE OF INJURY* _____
14. DUTY WHEN INJURY OCCURRED? *Press <Return> for EXPLAIN INJURY* _____
15. LOCATION WHERE INJURY OCCURRED _____
16. LOCATION WHERE DEATH OCCURRED _____
17. IMMEDIATE CAUSE OF DEATH _____
18. EMPLOYEE'S PAY RATE AS OF _____

	A. BASE PAY	B. SUBSISTENCE	C. QUARTERS	D. OTHER PAY
A. INJURY	\$ _____	\$ _____	\$ _____	\$ _____
	PER _____	PER _____	PER _____	PER _____

B. PAY STOPPED	\$ _____	\$ _____	\$ _____	\$ _____
	PER _____	PER _____	PER _____	PER _____

19. DID EMPLOYEE WORK IN POSITION FOR FULL 11 MONTHS? _____

20. WOULD POSITION HAVE AFFORDED EMPLOYMENT FOR ELEVEN MONTHS EXCEPT FOR THE INJURY? _____

21. LEAVE PAY RECEIVED, FROM: _____ THROUGH: _____

22. A. OCCUPATION CODE _____
B. TYPE CODE _____
C. SOURCE CODE _____
OWCP USE-NOI CODE _____

23. WAS CONTINUATION IF PAY (COP) RECEIVED? _____

A. PAY RATE USED FOR COP \$ _____ PER _____
TOTAL COP RECEIVED \$ _____

B. INCLUSIVE DATES OF COP, FROM: _____ TO: _____

24. HEALTH BENEFIT (HBS) CODE, IF ENROLLED _____

25. SHOW DATE THROUGH WHICH HBS LAST MADE _____

26. ATTENDING PHYSICIAN _____

ADDRESS _____

CITY _____ STATE _____ ZIP CODE _____

27. WAS INJURY CAUSED BY THIRD PARTY? _____

Press <Return> for NAME/ADDRESS THIRD PARTY

28. NAME OF ATTORNEY _____

ADDRESS _____

CITY _____ STATE _____ ZIP CODE _____

CA-6 CON'T

30. SHOW BRANCH OF SERVICE _____
SERIAL NUMBER (IF KNOWN) _____
31. HAS CLAIM FILE WITH OPM? _____
32. NAME OF SPOUSE OR NEXT OF KIN _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
34. SUPERVISOR'S TITLE _____
35. DATE SUPERVISOR SIGNED _____

THIRD PARTY INFORMATION

NAME _____ SETTLEMENT AMOUNT _____

ADDRESS _____

CITY _____ STATE _____ ZIP CODE _____

*Word Processing Window will appear.

M/FORM: CA-6 (#658)

I. PRE-FORM CODE

```
n file,nptfile,filing s:'$d(nocomp) file=~ptc(ifom,99)
+s:$d(nocomp) file=~ptg(903,ifom,99) s New=0,nptfile=1,ident=2 k ident
s U="" i '$d(DT) s X=$d($H),DT="" d ~DT s DT=Y k X,Y
S:$D(DA) $P(~ICC(DA,221),U,2,3)="" d sec~fmfrm
n file,nptfile,filing s:'$d(nocomp) file=~ptc(ifom,99)
```

II. WINDOWS

WL CAG

DATA ELEMENT	DATA TYPE	FOINDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	FOINDER 1004'	EMPLOY(
--------	---------------	---------

M/FORM FID # 70

```
17 EXTERNAL DEFAULT: NAME
18 INTERNAL DEFAULT: NAMEPTR
20 CONVERSION CODE: s cm=$e(cm:"",',($d(~EMPLOY(cm,0))#10):cm,1:$p(~EMPLOY(cm,0),"",1))
28 STORED: $p(~ICC(DA,0),"",3)
32 POST FILING CODE: d setindx~fmfrm
44 POST ID: I (new) s:'$d(~ICC("C",new,x1)) cm="",y=24,msg="CLAIM NOT LOGGED!"
54 INDEX: ~ICC("C")
55 LOOKUP NODE:
70~30~$p(~ICC(Lids,0),"",3)
8~23~$p(~ICC(Lids,0),"",2)
```

6 BIRTHDATE	DATE	S DT="E" D ~DT S X=Y K:Y<1 X
-------------	------	------------------------------

M/FORM FID # 2

```
20 CONVERSION CODE:
s X="(" s:~$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:~$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " $e(cm "000",9,10) _:" $e(cm "000",11,12) s X=X "_"
s:X="(1700)" X="" s cm=X
28 STORED: $p(~ICC(DA,0),"",7)
```

4 SEX	SET OF CODES	F:FEMALE M:MALE U:UNKNOWN
-------	--------------	---------------------------------

M/FORM FID # 3

```
20 CONVERSION CODE: s cm=$p($p("F:FEMALE;M:MALE;U:UNKNOWN;";";" _cm _:"",2);";",1),cm=$p(cm,"",1)
28 STORED: $p(~ICC(DA,0),"",5)
```

3 SSN

FREE TEXT

K:\$L(X)>9!(\$L(X)<9)!'(XZ9N) X

M/FORM FID # 4

28 STORED: \$p('ICC(DA,0),"",4)

44 POST ID: s %cm K:\$L(X)>9!(\$L(X)<9)!'(XZ9N) X s:'\$d(X) cm="" s:cm["" cm=""

13 DUTY STATION

FOINDER 1062' INJ(1062,

M/FORM FID # 38

20 CONVERSION CODE: s cm-\$s(cm="": "", '(\$d('INJ(1062,cm,0))#10):cm,1:\$p('INJ(1062,cm,0),"",1))

28 STORED: \$p('ICC(DA,9),"",1)

13.4 DUTY STATION ADDRESS

FREE TEXT

K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 39

28 STORED: \$p('ICC(DA,9),"",5)

44 POST ID: s %cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.5 DUTY STATION CITY

FREE TEXT

K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 40

28 STORED: \$p('ICC(DA,9),"",6)

44 POST ID: s %cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.6 DUTY STATION STATE

FOINDER 1003' DIZ(1003,

M/FORM FID # 41

20 CONVERSION CODE: s cm-\$s(cm="": "", '(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))

28 STORED: \$p('ICC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 42

28 STORED: \$p('ICC(DA,9),"",8)

44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

13.1 OACP AGENCY CODE

FREE TEXT

K:\$L(X)>6!(\$L(X)<6) X

M/FORM FID # 68

28 STORED: \$p('ICC(DA,9),"",2)

44 POST ID: s %cm K:\$L(X)>6!(\$L(X)<6) X s:'\$d(X) cm="" s:cm["" cm=""

13.9 OSHA STATE CODE

FREE TEXT

K:\$L(X)>9!(\$L(X)<1) X

M/FORM FID # 69

28 STORED: \$p('ICC(DA,9),"",10)

44 POST ID: s X=cm K:SL(X)>9!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

14 REPORTING OFFICE POINDEX 1063' INJ(1063,

M/FORM FID # 43

20 CONVERSION CODE: s cm=\$s(cm="" : "" , '(\$d(^INJ(1063,cm,0))#10):cm,1:\$p(^INJ(1063,cm,0), "" ,1))
28 STORED: \$p(^ICC(DA,10), "" ,1)

14.01 FO ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 44

28 STORED: \$p(^ICC(DA,10), "" ,2)
44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

14.02 FO CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 45

28 STORED: \$p(^ICC(DA,10), "" ,3)
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

14.03 FO STATE POINDEX 1003' DEZ(1003,

M/FORM FID # 46

20 CONVERSION CODE: s cm=\$s(cm="" : "" , '(\$d(^DEZ(1003,cm,0))#10):cm,1:\$p(^DEZ(1003,cm,0), "" ,1))
28 STORED: \$p(^ICC(DA,10), "" ,4)

14.04 FO ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 47

28 STORED: \$p(^ICC(DA,10), "" ,5)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

402 SUPERVISOR'S NAME FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 48

28 STORED: \$p(^ICC(DA,402), "" ,1)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

402.2 SUPERVISOR'S PHONE NUMBER FREE TEXT K:SL(X)>14!(SL(X)<7) X

M/FORM FID # 49

28 STORED: \$p(^ICC(DA,402), "" ,3)
44 POST ID: s X=cm K:SL(X)>14!(SL(X)<7) X s:'\$d(X) cm="" s:cm("" cm=""

1 INQUIRY/ILLNESS DATE & TI DATE S NOT="ET" D ^NOT S X=Y K:Y<1 X I \$d(X) K:\$p(X, "" ,1)>DT X

M/FORM FID # 8

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: Sp("ICC(DA,0),"",2)

32 POST FILING CODE: d setindx'fmr

54 INDEX: "ICC("D")

55 LOOKUP NOTE:

8"23~Sp("ICC(Lids,0),"",2)

70"30~Sp("ICC(Lids,0),"",3)

103 DATE OF DEATH

DATE

S NOT="ET" D NOT S X=Y K:Y<1 X I \$D(X) K:X~Sp("ICC(DA,0),U,2) X

M/FORM FID # 9

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: Sp("ICC(DA,100),"",4)

230 D/T BAY STOPPED

DATE

S NOT="ET" D NOT S X=Y K:Y<1 X

M/FORM FID # 10

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " \$e(cm "000",9,10) ":" \$e(cm "000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: Sp("ICC(DA,221),"",2)

120.1 CAUSE OF INJURY EXPLAINA

M/FORM FID # 11

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d filecns'ptrfilel k shhfile
i \$d(file),file,New=0 s filing=\$e(' \$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr'fmr s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: CAUSE OF INJURY (#641)

M/FORM: CAUSE OF INJURY (#641)

I. PRE-FORM CODE

```
n file,nptfile,New s:'$d(nocamp) file=~ptc(ifom,99)
+s:$d(nocamp) file=~ptg(903,ifom,99) s New=0,nptfile=1
s U="" i '$d(DT) s X=$d($H),WDT="" d ~WDT s DT=X k X,Y
d sec~fmfm
d crindx~fmfm
```

II. WINDOWS

CAUSE OF INJURY Window I

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 CAUSE OF INJURY EXPLAINA	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(~ICC(DA(1),111,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmfm d node0~fmfm
```

307 INJURED IN LINE OF DUTY SET OF CODES Y:YES
N:NO

M/FORM FID # 12

20 CONVERSION CODE: s am=\$p(\$p("Y:YES;N:NO;",";","_am_":",2)",";"),1),am=\$p(am,"",1)
28 STORED: \$p(~ICC(DA,300),"",8)

307.1 EXPLAIN INJURY

M/FORM FID # 13

28 STORED: nowhere

29 PREFIELD CODE:

```
i '$d(cbi(12)) s skip="D"
s:cbi(12)["Y"]!(cbi(12))["Y"] skip="D"
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileam~ptrfilel k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
```

```
s lvl=lvl+1
i lvl s DA(0)=DA,LO-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: EXPLAIN INQUIRY (#616)

M/FORM: EXPLAIN INQUIRY (#616)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=^ptc(iform,99)
+s:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),DT="" d ~DT s DT=Y k X,Y
d sec^fmfm
d crink^fmfm
```

II. WINDOWS

EXPLAIN INQUIRY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN INQUIRY	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),306,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

114 SPECIFIC LOCATION	FREE TEXT	K:\$L(X)>401(\$L(X)<1) X
M/FORM FID # 14		

M/FORM FID # 14

28 STORED: \$p(^ICC(DA,100),"",15)

44 POST ID: s X=on K:\$L(X)>401(\$L(X)<1) X s:\$d(X) on="" s:on["" on=""

612 LOCATION OF DEATH	FREE TEXT	K:\$L(X)>301(\$L(X)<1) X
M/FORM FID # 15		

M/FORM FID # 15

28 STORED: \$p(^ICC(DA,600),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

603 CAUSE OF DEATH FREE TEXT K:SL(X)>30!(SL(X)<3) X

M/FORM FID # 16

28 STORED: \$p(^ICC(DA,600), "",4)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm("" cm=""

W2 PAY INFORMATION

DATA ELEMENT DATA TYPE POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
21 PAY RATE NUMERIC FORMAT S:X("S" X=\$P(X,"S",2) K:+X'=X(X'?.N1". "2N)!(X>999999)!(X<0) X

M/FORM FID # 71

28 STORED: \$p(^ICC(DA,3), "",8)

20 PAY BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-WEEKLY
A:ANNUALLY
D:DAILY

M/FORM FID # 72

20 CONVERSION CODE: s cm=\$p(\$p(" ;H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;A:ANNUALLY;
D:DAILY;";", " cm ":"",2),";",1),cm=\$p(cm,";",1)
28 STORED: \$p(^ICC(DA,3), "",7)

21.1 DOI SUBSISTENCE NUMERIC FORMAT S:X("S" X=\$P(X,"S",2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 78

28 STORED: \$p(^ICC(DA,3), "",16)

21.25 DOI SUBSISTENCE BASIS SET OF CODES H:HOURLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 79

20 CONVERSION CODE: s cm=\$p(\$p(" ;H:HOURLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";", " cm ":"",2),";",1),cm=\$p(cm,";",1)
28 STORED: \$p(^ICC(DA,3), "",17)

21.3 DOI QUARTERS NUMERIC FORMAT S:X("S" X=\$P(X,"S",2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 75

28 STORED: \$p(ICC(DA,3),"",18)

21.35 DOI QUARTERS BASIS

SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 76

20 CONVERSION CODE: s cm \$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;"",") cm ":",2),";",1),cm \$p(cm,"",1)
28 STORED: \$p(ICC(DA,3),"",19)

21.4 DOI OTHER RATE

NUMERIC FORMAT S:X "\$" X-\$p(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 73

28 STORED: \$p(ICC(DA,3),"",20)

21.45 DOI OTHER RATE BASIS

SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 74

20 CONVERSION CODE: s cm \$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;"",") cm ":",2),";",1),cm \$p(cm,"",1)
28 STORED: \$p(ICC(DA,3),"",21)

229.1 WORK STOPPED BASE RATE

NUMERIC FORMAT S:X "\$" X-\$p(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 81

28 STORED: \$p(ICC(DA,22),"",1)

229.2 WORK STOPPED BASE BASIS

SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 80

20 CONVERSION CODE: s cm \$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;"",") cm ":",2),";",1),cm \$p(cm,"",1)
28 STORED: \$p(ICC(DA,22),"",2)

229.3 WORK STOPPED SUBSISTENCE

NUMERIC FORMAT S:X "\$" X-\$p(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 83

28 STORED: \$p(ICC(DA,222),"",3)

229.35 WORK STOPPED SUB BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 82

20 CONVERSION CODE: s cm \$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm \$p(cm,":",1)
28 STORED: \$p(ICC(DA,222),"",4)

229.4 WORK STOPPED QUARTERS NUMERIC FORMAT S:X["\$" X=\$p(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 84

28 STORED: \$p(ICC(DA,222),"",5)

229.45 WORK STOPPED QUARTERS BA SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 85

20 CONVERSION CODE: s cm \$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm \$p(cm,":",1)
28 STORED: \$p(ICC(DA,222),"",6)

229.5 WORK STOPPED OTHER RATE NUMERIC FORMAT S:X["\$" X=\$p(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 87

28 STORED: \$p(ICC(DA,222),"",7)

229.55 WORK STOPPED OTHER BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 86

20 CONVERSION CODE: s cm \$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;A:ANNUALLY;
D:DAILY;";";" cm ":"2);";1),cm \$p(cm,":",1)
28 STORED: \$p(ICC(DA,222),"",8)

W3 C05

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
605 EMPLOYED IN DUTY 11 MOS	SET OF CODES	Y:YES N:NO

M/FORM FID # 18

20 CONVERSION CODE: s cm=Sp(\$p("Y:YES;N:NO;"_"cm":",2),"_",1),cm=Sp(cm,"_",1)
28 STORED: Sp("ICC(DA,600),"",2)

606 DUTY AFFORDED 11 MOS WOR	SET OF CODES	Y:YES N:NO
------------------------------	--------------	---------------

M/FORM FID # 19

20 CONVERSION CODE: s cm=Sp(\$p("Y:YES;N:NO;"_"cm":",2),"_",1),cm=Sp(cm,"_",1)
28 STORED: Sp("ICC(DA,600),"",3)
29 PREFIELD CODE:
I 'SD(chi(18)) s skip="D",chi(0,19)=""
i \$d(chi(18)) s:chi(18){"Y"!(chi(18){"Y"}) skip="D",chi(0,19)=""

251 OTHER LEAVE FROM	DATE	S %DT="E" D %DT S %Y K:Y<1 X
----------------------	------	------------------------------

M/FORM FID # 88

20 CONVERSION CODE:
s %="(" s:+\$e(cm,4,5) %K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","_",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,"_",2)'="" %K " at " \$e(cm "000",9,10) " " \$e(cm "000",11,12) s %K ")"
s:%="(1700)" %="" s cm=K

28 STORED: Sp("ICC(DA,250),"",7)

15.5 OCCUPATION CODE	FREE TEXT	K:SL(X)>6!(SL(X)<5) X
----------------------	-----------	-----------------------

M/FORM FID # 21

28 STORED: Sp("ICC(DA,3),"",2)
44 POST ID: s %cm K:SL(X)>6!(SL(X)<5) X s:\$d(X) cm="" s:cm["" cm=""

251.5 OTHER LEAVE THRU	DATE	S %DT="E" D %DT S %Y K:Y<1 X
------------------------	------	------------------------------

M/FORM FID # 89

20 CONVERSION CODE:
s %="(" s:+\$e(cm,4,5) %K Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","_",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) " " s %K (1700+\$e(cm,1,3))
s:\$p(cm,"_",2)'="" %K " at " \$e(cm "000",9,10) " " \$e(cm "000",11,12) s %K ")"
s:%="(1700)" %="" s cm=K

28 STORED: \$p(^IOC(DA,250),"",8)
 29 PREFIELD CODE: S:\$D(cbi(88)) skip="D"

112.5 TYPE CODE FREE TEXT K:\$L(X)>5!(\$L(X)<1) X

M/FORM FID # 22

28 STORED: \$p(^IOC(DA,100),"",17)
 44 POST ID: s %cm K:\$L(X)>5!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

126.5 SOURCE CODE FREE TEXT K:\$L(X)>6!(\$L(X)<1) X

M/FORM FID # 23

28 STORED: \$p(^IOC(DA,120),"",9)
 44 POST ID: s %cm K:\$L(X)>6!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

101.2 OACP USE-NOI CODE FREE TEXT K:\$L(X)>5!(\$L(X)<1) X

M/FORM FID # 24

28 STORED: \$p(^IOC(DA,100),"",18)
 44 POST ID: s %cm K:\$L(X)>5!(\$L(X)<1) X s:\$d(X) cm="" s:cm["" cm=""

250.7 COP RATE NUMERIC FORMAT S:X["\$"] X=\$P(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 92

28 STORED: \$p(^IOC(DA,250),"",17)

250.8 COP RATE BASIS SET OF CODES
 H:HOURLY
 W:WEEKLY
 M:MONTHLY
 B:BI-WEEKLY
 A:ANNUALLY
 D:DAILY

M/FORM FID # 93

20 CONVERSION CODE: s cm=\$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;A:ANNUALLY;D:DAILY;"",",) cm ":",2),",",1),cm=\$p(cm,"",1)
 28 STORED: \$p(^IOC(DA,250),"",18)
 29 PREFIELD CODE: S:\$D(cbi(92)) skip="D"

250.6 COP RECD GROSS NUMERIC FORMAT S:X["\$"] X=\$P(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 94

28 STORED: \$p(^IOC(DA,250),"",16)
 29 PREFIELD CODE: S:\$D(cbi(92)) skip="D"

250 COP RECD FROM DATE S %DT="E" D %DT S %Y K:Y<1 X

M/FORM FID # 90

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",1)

250.5 COP RECVD THRU DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 91

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",2)

29 PREFIELD CODE: S:'SD(chi(90)) skip="D"

25 FEHB CODE FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 25

28 STORED: Sp(^ICC(DA,3),"",13)

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

332 DATE HEALTH DEDUCT STOPP DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 34

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,303),"",12)

400 ATTENDING PHYSICIAN FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 52

28 STORED: Sp(^ICC(DA,400),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

400.1 DOCS ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 53

28 STORED: Sp(^ICC(DA,400),"",2)

44 POST ID: s X=cm K:SL(X)>45!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

400.2 DOCS CITY

FREE TEXT

K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 54

28 STORED: \$p(^ICC(DA,400),"",3)

44 POST ID: s %cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.3 DOCS STATE

FOINDER 1003' DIZ(1003,

M/FORM FID # 55

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,400),"",4)

400.4 DOCS ZIPCODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 56

28 STORED: \$p(^ICC(DA,400),"",5)

44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

WJ 046

DATA ELEMENT

DATA TYPE

FOINDER GLOBAL REFERENCE OR INPUT VERIFICATION

140 THIRD PARTY CASE

SET OF CODES

Y:YES

N:NO

M/FORM FID # 35

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",",",2),",",1),cm=\$p(cm,"",1)

28 STORED: \$p(^ICC(DA,130),"",9)

141.1 THIRD PARTY

M/FORM FID # 27

28 STORED: nowhere

29 PREFIELD CODE:

i '\$d(cbi(35)) s skip="D"

s:cbi(35){ "n"! (cbi(35){ "N") skip="D"

s:'\$d(lvl) lvl=0 i '\$d(cbi(0))>9 s shifile=l d filecm=ptrfilel k shifile

i '\$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)

+d upcntr^fufm s New=0,cbi(0)="" k file

s lvl=lvl+1

i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)

s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

16 CALLS M/FORM: NAME/ADDRESS THIRD PARTY (#626)

M/FORM: NAME/ADDRESS THIRD PARTY (#626)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=^ptc(iform,99)
+s:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U=" " i '$d(DT) s X=$d($H),%DT=" " d ^DT s DT=Y k X,Y
d sec^fmfm
d crinck^fmfm
```

II. WINDOWS

NAME/ADDRESS THIRD PARTY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
----------	-----------	--------------------------

M/FORM FID # 6

28 STORED: \$p(^ICC(DA(1),140,DA,0),"",1)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

5 SETTLEMENT AMOUNT	NUMERIC FORMAT	S:X' "\$" X=\$P(X,"\$",2) K:+X'-X(X'?.N1". "2N)!(X>99999999.99)!(X<0) X
---------------------	----------------	---

M/FORM FID # 7

28 STORED: \$p(^ICC(DA(1),140,DA,0),"",6)

1 ADDRESS	FREE TEXT	K:\$L(X)>70!(\$L(X)<1) X
-----------	-----------	--------------------------

M/FORM FID # 2

28 STORED: \$p(^ICC(DA(1),140,DA,0),"",2)
44 POST ID: s X=cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

2 CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
--------	-----------	--------------------------

M/FORM FID # 3

28 STORED: \$p(^ICC(DA(1),140,DA,0),"",3)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

3 STATE	POINTER 1003'	DIZ(1003,
---------	---------------	-----------

M/FORM FID # 4

20 CONVERSION CODE: s cm=\$s(cm="":'(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^ICC(DA(1),140,DA,0),"",4)

4 ZIPCODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
-----------	-----------	--------------------------

M/FORM FID # 5

28 STORED: \$p(^IOC(DA(1),140,DA,0),"",5)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0^fmfm

609 ATTORNEY NAME FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 59

28 STORED: \$p(^IOC(DA,601),"",1)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

609.1 ATTORNEY ADDRESS FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 57

28 STORED: \$p(^IOC(DA,601),"",2)
44 POST ID: s X=cm K:\$L(X)>54!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

609.2 ATTORNEY CITY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 58

28 STORED: \$p(^IOC(DA,601),"",3)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

609.3 ATTORNEY STATE FOUNDER 1003' DIEZ(1003,

M/FORM FID # 60

20 CONVERSION CODE: s cm-\$s(cm="": "",('\$d(^DIEZ(1003,cm,0))%13):cm,1:\$p(^DIEZ(1003,cm,0),"",1))
28 STORED: \$p(^IOC(DA,601),"",4)

609.4 ATTORNEY ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 61

28 STORED: \$p(^IOC(DA,601),"",5)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

506 BRANCH OF SERVICE SET OF CODES A:ARMY
N:NAVY
M:MARINES
F:AIR FORCE
C:COAST GUARD

M/FORM FID # 30

28 STORED: \$p(^ICC(DA,500),"",4)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

505 SERVICE NO FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 31

28 STORED: \$p(^ICC(DA,500),"",3)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

610 CLAIM FILED W/CM SET OF CODES Y:YES
N:NO

M/FORM FID # 32

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA,600),"",5)

611 NEXT OF KIN FREE TEXT K:\$L(X)>35!(\$L(X)<1) X

M/FORM FID # 62

28 STORED: \$p(^ICC(DA,602),"",1)
44 POST ID: s X=cm K:\$L(X)>35!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

611.1 ADDRESS OF NEXT OF KIN FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 63

28 STORED: \$p(^ICC(DA,602),"",2)
44 POST ID: s X=cm K:\$L(X)>54!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

611.2 CITY OF NEXT OF KIN FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 64

28 STORED: \$p(^ICC(DA,602),"",3)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

611.3 STATE OF NEXT OF KIN POINTER 1003' DIZ(1003,

M/FORM FID # 65

20 CONVERSION CODE: s cm=\$s(cm="":""',('\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^ICC(DA,602),"",4)

611.4 ZIP CODE OF NEXT OF KIN FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 66

28 STORED: \$p(^ICC(DA,602),"",5)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

402.1 SUPERVISOR'S TITLE FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 50

28 STORED: \$p(^ICC(DA,402),"" ,2)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

402.4 DATE SUPERVISOR SIGNED DATE S %DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 51

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " , " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm "000",9,10) _:" \$e(cm "000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,402),"" ,5)

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fncm
i file=1!(\$d(ERFLAG)) D C06^T3CPNF, ^T3CTLE

CA-7 CLAIM FOR COMPENSATION
EMPLOYEE STATEMENT

1. NAME OF EMPLOYEE _____
2. OWCP FILE NUMBER _____ 3. SOCIAL SECURITY NUMBER _____
4. PERIOD OF WAGE LOSS, FROM _____ THROUGH _____ HRS _____
5. IS THIS A CLAIM FOR A SCHEDULE AWARD? _____
6. ANY PAY RECEIVED FOR PERIOD IN ITEM #4? _____
7. IF YES, AMOUNT \$ _____
COMPENSATION RECEIVED FROM: _____ THROUGH: _____
8. WAS CLAIM AGAINST A THIRD PARTY? _____
9. NAME OF THIRD PARTY/INSURANCE CARRIER _____
Press <Return> for NAME/ADDRESS THIRD PARTY
10. HAS CASE BEEN SETTLED? _____

EMPLOYEE STATEMENT PAGE 2

11. APPLIED OR RECEIVED BENEFITS FROM VA? _____
 - A. CLAIM NUMBER _____
 - B. VA OFFICE WHERE CLAIM FILED _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
 - C. NATURE OF DISABILITY _____
MONTHLY PAYMENTS \$ _____
12. APPLIED OR RECEIVED AN ANNUITY? _____
 - A. CLAIM NUMBER _____
 - B. DATE ANNUITY BEGAN _____
 - C. AMOUNT OF MONTHLY PAYMENT \$ _____

DEPENDENTS

- 13-15, 17. LIST YOUR DEPENDENTS Press <Return> for DEPENDENTS
16. *Press <Return> for SUPPORT PAID*

SIGNATURE OF EMPLOYEE

18. DATE EMPLOYEE SIGNED (MO., DAY, YEAR) _____
19. EMPLOYEE'S HOME MAILING ADDRESS _____
STREET _____
CITY _____ STATE _____ ZIP CODE _____
OVERSEAS _____

STATEMENT OF OFFICIAL SUPERIOR

20. RATE AS OF A. BASE PAY B. SUBSISTENCE C. QUARTERS D. OTHER
- A. INJURY \$ _____ \$ _____ \$ _____ \$ _____
 PER _____ PER _____ PER _____ PER _____
- B. STOPPED \$ _____ \$ _____ \$ _____ \$ _____
 WORK PER _____ PER _____ PER _____ PER _____
21. RECEIVED PAY, IDENTIFY TYPE AND AMOUNT
- PREMIUM PAY _____ PFR _____ NIGHT PAY _____ PER _____
 SUNDAY PAY _____ PER _____ OTHER PAY _____ PER _____
 IDENTIFY OTHER PAY _____
22. SHOW WORK SCHEDULE FOR WEEK PAY STOPPED _____
23. WORK IN POSITION 11 MOS PRIOR TO INJURY? _____
24. DUTY AFFORDED 11 MONTHS WORK? _____
25. TOTAL LENGTH FEDERAL CIVIL SERVICE, YRS. _____ MOS. _____

HEALTH BENEFITS AND OPTIONAL LIFE INSURANCE

26. ENROLLED IN HEALTH BENEFITS? _____
 IF YES, GIVE CODE _____
 ENDING DATE OF PAY PERIOD DEDUCTIONS MADE _____
27. ENROLLED IN OPTIONAL LIFE INSURANCE PRG? _____
 IF YES, WAS EMPLOYEE ENROLLED IN OPTION _____
 IF OPTION B, SHOW NUMBER OF MULTIPLES _____

LEAVE AND CONTINUATION OF PAY

28. SICK LEAVE RECEIVED, FROM: _____ THRU: _____
 ANNUAL LEAVE RECEIVED, FROM: _____ THRU: _____
 OTHER LEAVE RECEIVED, FROM: _____ THRU: _____
 OTHER LEAVE SPECIFY _____
29. CONTINUATION OF PAY, FROM: _____ THRU: _____
30. DATE ALL PAY STOPPED _____
31. PERIOD COMPENSATION IS CLAIMED, FROM: _____ THRU: _____

RETURN TO DUTY

32. DATE RETURNED TO WORK _____
33. WORK SCHEDULE WHEN RETURNED TO WORK _____
34. WORK ASSIGNMENT CHANGE? _____
 DESCRIBE NEW WORK ASSIGNMENT _____
 Press <Return> for DESCRIBE NEW DUTIES
35. PAY RATE ON RETURN TO WORK \$ _____ PER _____

CI-7 CON'T

CERTIFICATION

36. OFFICIAL REMARKS _____
SUPERVISOR _____ DATE SIGNED _____
SUPERVISOR'S TITLE _____
AGENCY NAME _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
37. PERSON CONTACT, IF OWCP NEEDS PAY INFO. _____
POINT OF CONTACT _____ PHONE _____

DEPENDENTS

NAME _____
RELATIONSHIP _____ BIRTHDATE _____
STREET ADDRESS _____ LIVING WITH YOU? _____
CITY _____ STATE _____ ZIP CODE _____
SUPPORT PAYMENTS? _____
ORDERED BY THE COURT? _____ AMOUNT OF PAYMENTS \$ _____ PER _____

*Word Processing Window will appear.

M/FORM: OA-7 (#671)

I. PRE-FORM CODE

```
n file, rptfile, filing s: 'Sd(nocomp) file=^ptc(iform,99)
+s: 'Sd(nocomp) file=^ptg(903,iform,99) s New=0, rptfile=1, idedist=2 k ident
s U=" " i 'Sd(DT) s X=Sd(JH), %DT=" " d ^DT s DT=Y k X,Y
I Sd(DA) S Sp(^ICC(DA,221),U,2,3)=" " d sec^mfmm
n file, rptfile, filing s: 'Sd(nocomp) file=^ptc(iform,99)
```

II. WINDOWS

WI EMPLOYEE STATEMENT

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	POINTER 1004'	EMPLOY(
M/FORM FID # 82		
17 EXTERNAL DEFAULT: NAME		
18 INTERNAL DEFAULT: NAMEPIR		
20 CONVERSION CODE: s cm=\$s(cm="": "", '(Sd(^EMPLOY(cm,0))#10):cm,1:Sp(^EMPLOY(cm,0),"",1))		
28 STORED: Sp(^ICC(DA,0),"",3)		
32 POST FILING CODE: d setindx^mfmm		
44 POST ID: i Sd(new) s: 'Sd(^ICC("C",new,x1)) msg="Claim not logged",y=25,cm=""		
54 INDEX: ^ICC("C")		
55 LOOKUP NODE: 82^45^Sp(^ICC(Lids,0),"",3)		
137 OACP FILE NUMBER	FREE TEXT	K:SL(X)>131(SL(X)<1) X
M/FORM FID # 2		
28 STORED: Sp(^ICC(DA,130),"",6)		
44 POST ID: s X=cm K:SL(X)>131(SL(X)<1) X s: 'Sd(X) cm=" " s:cm(" " cm=" "		
3 SSN	FREE TEXT	K:SL(X)>91(SL(X)<9)!(XG9N) X
M/FORM FID # 3		
28 STORED: Sp(^ICC(DA,0),"",4)		
44 POST ID: s X=cm K:SL(X)>91(SL(X)<9)!(XG9N) X s: 'Sd(X) cm=" " s:cm(" " cm=" "		
701 DATE WAGES LAST FROM	DATE	S ^DT="E" D ^DT S X=Y K:Y<1 X
M/FORM FID # 45		
20 CONVERSION CODE:		
s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",		
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))		
s:Sp(cm,".",2)'=" " X=X " at " _Se(cm^"000",9,10)'=" _Se(cm^"000",11,12) s X=X "		
s:X="(1700)" X=" " s cm=X		
28 STORED: Sp(^ICC(DA,700),"",7)		

701.5 DATES WAGES LOST THRU

DATE

S WDT="E" D WDT S X=Y K:Y<1 X

M/FORM FID # 46

20 CONVERSION CODE:

s X="(" s:+\$(cm,4,5) X=X \$(JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ".
+\$(cm,4,5)) " " s:+\$(cm,6,7) X=X \$(cm,6,7) " " s X=X (1700+\$(cm,1,3))
s:\$(cm,".",2)'=" X=X " at " \$(cm"000",9,10) ":" \$(cm"000",11,12) s X=X "
s:X="(1700)" X="" s cm=X

28 STORED: \$(^ICC(DA,700),""",9)

44 POST ID:

i \$(chi(45)) s skip="D"
s X=cm k:X<\$(chi(45),cb,2) X i '\$d(X) s cm="" s:cm[""" cm=""

702 HRS COMP CLAIMED

NUMERIC FORMAT K:+X'-X!(X>999)!(X<0)!(X?.E1". "N.N) X

M/FORM FID # 4

28 STORED: \$(^ICC(DA,700),""",2)

703 SCHEDULE AWARD

SET OF CODES

Y:YES

N:NO

M/FORM FID # 5

20 CONVERSION CODE: s cm=\$(\$(;"Y:YES;N:NO;" ";"_cm ":"",2) ";"",1),cm=\$(cm," ":"",1)

28 STORED: \$(^ICC(DA,700),""",3)

704 WAS ANY PAY RECEIVED

SET OF CODES

Y:YES

N:NO

M/FORM FID # 6

20 CONVERSION CODE: s cm=\$(\$(;"Y:YES;N:NO;" ";"_cm ":"",2) ";"",1),cm=\$(cm," ":"",1)

28 STORED: \$(^ICC(DA,700),""",4)

256.2 COMP RECD GROSS

NUMERIC FORMAT

S:X('\$' X=\$(X,"\$",2) K:+X'-X&(X?7.N1". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 129

28 STORED: \$(^ICC(DA,251),""",3)

29 PREFIELD CODE: S:'\$(chi(6)) skip="D" i '\$d(skip) s:chi(6)'["Y" skip="D"

256 COMP RECD FROM

DATE

S WDT="E" D WDT S X=Y K:Y<1 X

M/FORM FID # 127

20 CONVERSION CODE:

s X="(" s:+\$(cm,4,5) X=X \$(JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ".
+\$(cm,4,5)) " " s:+\$(cm,6,7) X=X \$(cm,6,7) " " s X=X (1700+\$(cm,1,3))
s:\$(cm,".",2)'=" X=X " at " \$(cm"000",9,10) ":" \$(cm"000",11,12) s X=X "
s:X="(1700)" X="" s cm=X

256.1 COMP RECVD THRU DATE S ^DT="E" D ~^DT S X=Y K:Y<1 X

d crink`fmfm

II. WINDOWS

NAME/ADDRESS THIRD PARTY Window 1

DATA ELEMENT	DATA TYPE	FOINIER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 6		
28 STORED: \$p(`IOC(DA(1),140,DA,0),"",1)		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""		
5 SETTLEMENT AMOUNT	NUMERIC FORMAT	S:X["\$"] X=\$P(X,"\$",2) K:+K'=X(X'?N1"."2N)!(X>99999999.99)!(X<0) X
M/FORM FID # 7		
28 STORED: \$p(`IOC(DA(1),140,DA,0),"",6)		
1 ADDRESS	FREE TEXT	K:\$L(X)>70!(\$L(X)<1) X
M/FORM FID # 2		
28 STORED: \$p(`IOC(DA(1),140,DA,0),"",2)		
44 POST ID: s X=cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""		
2 CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
M/FORM FID # 3		
28 STORED: \$p(`IOC(DA(1),140,DA,0),"",3)		
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""		
3 STATE	FOINIER 1003' DIZ(1003,	
M/FORM FID # 4		
20 CONVERSION CODE: s cm=\$s(cm=""":'(\$d(`DIZ(1003,cm,0))#10):cm,1:\$p(`DIZ(1003,cm,0),"",1))		
28 STORED: \$p(`IOC(DA(1),140,DA,0),"",4)		
4 ZIPCODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
M/FORM FID # 5		
28 STORED: \$p(`IOC(DA(1),140,DA,0),"",5)		
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm[""] cm=""		

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0`fmfm

141 CASE OUTCOME	FREE TEXT	K:\$L(X)>60!(\$L(X)<1) X
------------------	-----------	--------------------------

M/FORM FID # 9

28 STORED: \$p(^IOC(DA,130),"^",10)

29 PREFIELD CODE:

```
I '$D(cbi(7)) s skip="D",cbi(0,9)=""
```

```
i $d(cbi(7)) s:cbi(7){"n"!(cbi(7){"N"} skip="D",cbi(0,9)=""
```

44 POST ID: s X=on K:\$L(X)>60!(\$L(X)<1) X s:'\$d(X) on="" s:an("" on=""

W2 ANNUITY

DATA ELEMENT

DATA TYPE

FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION

708 APPLY/RECVD VA BENEFITS

SET OF CODES

Y: YES

N:NO

M/FORM FID # 10

20 CONVERSION CODE: ~~S~~ ~~am~~~~Sp~~(~~Sp~~(";Y:YES;N:NO;"",";" ~~am~~ ":"",2),";",1),~~am~~~~Sp~~(~~am~~ ":"",1)

28 STORED: \$0(^ICC(DA,700),"^",8)

710 VA CLAIM NUMBER

FREE TEXT

R:SL(X)>15!(SL(X)<1) X

M/FORM FID # 66

28 STORED: \$p(^IOC(DA,701),"^",1)

29 PREFIELD CODE:

```
i '$d(cbi(10)) s skip="D"
```

```
i $d(obi(10)) s:obi(10)'{"Y" skip="D"
```

44 POST ID: s X-on K:SL(X)>15!(SL(X)<1) X s:'Sd(X) on"" s:om'"" on""

710.1 VA OFFICE WHERE CLAIM FI

FREE TEXT

$$R: SL(X) > 30 \mid (SL(X) < 1) \quad X$$

M/FORM FID # 67

28 STORED: \$p(^ICC(DA,701),"^",2)

29 PREFIELD CODE:

```
i '$(abi(10)) s skip="D"
```

```
i $d(obi(10)) s:obi(10)'["Y" skip="D"
```

44 POST ID: s %om K:SL(X)>30!(SL(X)<1) X s:'Sd(X) om"" s:om"" om""

710.2 VA STREET ADDRESS

FREE TEST

R:SL(X)>54!(SL(X)<1) X

M/FORM FID # 69

28 STORED: \$D(^IOC(DA,701),"",3)

29 PREFIELD CODE:

```
i 'Sd(cbi(10)) s skip="D"
```

```
i $d(obi(10)) s:obi(10)'["Y" skip="D"
```

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710.3 VA CITY

FREE TEXT

K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 65

28 STORED: Sp(^ICC(DA,701),"",4)

29 PREFIELD CODE:

i '\$d(cbi(10)) s skip="D"

i '\$d(cbi(10)) s:cbi(10)'["Y" skip="D"

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710.4 VA STATE

FOUNDER 1003' DIZ(1003,

M/FORM FID # 68

20 CONVERSION CODE: s cm=\$s(cm="": "",('\$d(^DIZ(1003,cm,0))#10):cm,1:Sp(^DIZ(1003,cm,0),"",1))

28 STORED: Sp(^ICC(DA,701),"",5)

29 PREFIELD CODE:

i '\$d(cbi(10)) s skip="D"

i '\$d(cbi(10)) s:cbi(10)'["Y" skip="D"

710.5 VA ZIP CODE

FREE TEXT

K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 70

28 STORED: Sp(^ICC(DA,701),"",6)

29 PREFIELD CODE:

i '\$d(cbi(10)) s skip="D"

i '\$d(cbi(10)) s:cbi(10)'["Y" skip="D"

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

710.6 VA NATURE OF DISABILITY

FREE TEXT

K:SL(X)>57!(SL(X)<1) X

M/FORM FID # 63

28 STORED: Sp(^ICC(DA,701),"",7)

29 PREFIELD CODE:

i '\$d(cbi(10)) s skip="D"

i '\$d(cbi(10)) s:cbi(10)'["Y" skip="D"

44 POST ID: s X=cm K:SL(X)>57!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

710.7 VA AMOUNT OF PAYMENT

NUMERIC FORMAT

S:X["\$" X=Sp(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>99999.99)!(X<0) X

M/FORM FID # 64

28 STORED: Sp(^ICC(DA,701),"",8)

29 PREFIELD CODE:

i '\$d(cbi(10)) s skip="D"

i '\$d(cbi(10)) s:cbi(10)'["Y" skip="D"

713 ANY OTHER RETIRE/DISABIL SET OF CODES Y: YES
N: NO

M/FORM FID # 12

20 CONVERSION CODE: s cm=Sp(Sp(";Y: YES;N: NO;";";" _cm_":",2);";",1),cm=Sp(cm,";",1)
28 STORED: Sp(^ICC(DA,700),"^",10)

714 OTHER ANNUITIES CLAIM NU FREE TEXT K: \$L(X)>15!(\$L(X)<1) X

M/FORM FID # 71

28 STORED: Sp(^ICC(DA,707)," ",1)
29 PREFIELD CODE:
i ' \$d(cbi(12)) s skip="D"
i \$d(cbi(12)) s: cbi(12) ["Y" skip="D"

44 POST ID: s X=cm K: \$L(X)>15!(\$L(X)<1) X s: ' \$d(X) cm="" s: cm("" cm=""

714.1 OTHER ANNUITY DATE BEGAN DATE S NOT="E" D NOT S X=Y K: Y<1 X

M/FORM FID # 74

20 CONVERSION CODE:
s X="(" s: +Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ". "
+Se(cm,4,5)) " " s: +Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s: Sp(cm,".",2)'="" X=X " at " _Se(cm"000",5,10) _": " _Se(cm"000",11,12) s X=X _")"
s: X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,707),"",2)
29 PREFIELD CODE:
i ' \$d(cbi(12)) s skip="D"
i \$d(cbi(12)) s: cbi(12) ["Y" skip="D"

714.2 OTHER ANNUITY AMOUNT NUMERIC FORamt S: X("\$" X=Sp(X,"\$",2) K: +X'=X(X' ? .NL". "2N)!(X>99999999.99)!(X<0) X

M/FORM FID # 72

28 STORED: Sp(^ICC(DA,707),"",3)
29 PREFIELD CODE:
i ' \$d(cbi(12)) s skip="D"
i \$d(cbi(12)) s: cbi(12) ["Y" skip="D"

W3 DEPENDENTS

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

722 DEPENDENT INFO

M/FORM FID # 14

28 STORED: nowhere
29 PREFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone~ptrfile1 k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

```

16 CALLS M/FORM: DEPENDENTIS (#686)

M/FORM: DEPENDENTIS (#686)

I. PRE-FORM CODE

```

n file,rptfile,New s:'$d(nocomp) filen=~ptc(iform,99)
+s:$d(nocomp) filen=~ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),DT="" d ~DT s DT=Y k X,Y
d sec~fmfm

```

II. WINDOWS

DEPENDENTIS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 1

```

28 STORED: $p(~IOC(DA(1),706,DA,0),"",1)
32 POST FILING CODE: d setindx~fmfm
44 POST ID: s X=cm K:$L(X)>30!( $L(X)<1) X s:$d(X) cm="" s:cm( "" cm=""
54 INDEX: ~IOC(DA(1),706,"B")
55 LOOKUP NODE: 1~30~$p(~IOC(DA(1),706,lids,0),"",1)

```

1 BIRTHDATE	DATE	S ~DT="E" D ~DT S X=Y K:Y<1 X
-------------	------	-------------------------------

M/FORM FID # 2

20 CONVERSION CODE:

```

s X="(" s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " $e(cm"000",9,10) _:" $e(cm"000",11,12) s X=X _"
s:X="(1700)" X="" s cm=X

```

```

28 STORED: $p(~IOC(DA(1),706,DA,0),"",2)

```

2 RELATIONSHIP	FREE TEXT	K:\$L(X)>15!(\$L(X)<1) X
----------------	-----------	---------------------------

M/FORM FID # 3

```

28 STORED: $p(~IOC(DA(1),706,DA,0),"",3)

```

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

3 LIVING WITH YOU SET OF CODES YES:YES
NO:NO

M/FORM FID # 4

20 CONVERSION CODE: s cm=\$p(\$p(";YES:YES;NO:NO;y:YES;n:NO;" ";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA(1),706,DA,0),"",4)

9 STREET ADDRESS FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 9

28 STORED: \$p(^ICC(DA(1),706,DA,0),"",5)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

10 CITY FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 10

28 STORED: \$p(^ICC(DA(1),706,DA,0),"",10)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

11 STATE FOURIER 1003' DIZ(1003,

M/FORM FID # 11

20 CONVERSION CODE: s cm=\$s(cm="" "";'(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))
28 STORED: \$p(^ICC(DA(1),706,DA,0),"",11)

12 ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 12

28 STORED: \$p(^ICC(DA(1),706,DA,0),"",12)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

5 SUPPORT PAYMENTS? SET OF CODES Y:YES
N:NO

M/FORM FID # 5

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;y:YES;n:NO;" ";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA(1),706,DA,0),"",6)

6 ORDERED BY THE COURT? SET OF CODES Y:YES
N:NO

M/FORM FID # 6

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;n:NO;y:YES;" ";"_cm_":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p(^ICC(DA(1),706,DA,0),"",7)
29 PREFIELD CODE:

7 AMOUNT OF PAYMENTS NUMERIC FORMAT S:X["\$"] *F(X,"\$",2) K:+X'=+X(X'? .N1". "2N)!(X>9999.99)!(X<0) X

```
i $d(obl(5)) s skip="D"
i $d(obl(5)) s:obl(5){ "N"!(obl(5){ "n" } skip="D"
```

SET OF CODES M:MONTHLY
W:WEEKLY
B:BI-WEEKLY
A:ANNUALLY

```
I 'Sd(cbi(5)) s skip="D"
i $d(cbi(5)) s:cbi(5){ "n"!(cbi(5){ "N") skip="D"
```

```
i $d(file),file s filing-$s('$(filing):lvl,lvl>filing:lvl,l:filing)
+d node0`fmfm
```

```
s: '$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone ptrfile k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr"fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="" DI=""
```

```

i lvl s DA(0)-DA f i=1:l:lvl s:$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),dl s lvl=lvl-1

```

M/FORM: SUPPORT BAJD (#732)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d %DT s DT=Y k X,Y
d sec^fmfm
```

II. WINDOWS

OTHER DEPENDENTS CA-7 Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<3) X
----------	-----------	--------------------------

M/FORM FID # 1

```
28 STORED: $p(^ICC(DA(1),519,DA,0),"",1)
32 POST FILING CODE: d setindx^fmfm
44 POST ID: s X=cm K:$L(X)>30!($L(X)<3) X s:'$d(X) cm="" s:cm["" cm=""
54 INDEX: ^ICC(DA(1),519,"B")
55 LOOKUP NODE: 1^30$P(^ICC(DA(1),519,1ids,0),"",1)
```

3 ADDRESS	FREE TEXT	K:\$L(X)>60!(\$L(X)<1) X
-----------	-----------	--------------------------

M/FORM FID # 2

```
28 STORED: $p(^ICC(DA(1),519,DA,0),"",4)
44 POST ID: s X=cm K:$L(X)>60!($L(X)<1) X s:'$d(X) cm="" s:cm["" cm=""
```

4 CITY	FREE TEXT	K:\$L(X)>23!(\$L(X)<1) X
--------	-----------	--------------------------

M/FORM FID # 3

```
28 STORED: $p(^ICC(DA(1),519,DA,0),"",5)
44 POST ID: s X=cm K:$L(X)>23!($L(X)<1) X s:'$d(X) cm="" s:cm["" cm=""
```

5 STATE	POINTER 1003' DIZ(1003,	
---------	-------------------------	--

M/FORM FID # 4

```
20 CONVERSION CODE: s cm=$s(cm="":,"',"($d(^DIZ(1003,cm,0))#10):cm,1:$p(^DIZ(1003,cm,0),"",1))
28 STORED: $p(^ICC(DA(1),519,DA,0),"",6)
```

6 ZIP CODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
------------	-----------	--------------------------

M/FORM FID # 5

```
28 STORED: $p(^ICC(DA(1),519,DA,0),"",7)
44 POST ID: s X=cm K:$L(X)>10!($L(X)<5) X s:'$d(X) cm="" s:cm["" cm=""
```

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0~fmfm

322 DATE EMP/CLAIMANT SIGNED DATE S %DT="E" D %DT S %Y K:Y<1 X

M/FORM FID # 18

20 CONVERSION CODE:

s %="(" s:+\$e(cm,4,5) %K \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %K \$e(cm,6,7) ", " s %K (1700+\$e(cm,1,3))
s:\$p(cm,".",2) %=" %X " at "\$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s %K "
s:%="(1700)" %=" s cm=X

28 STORED: \$p("%IC(DA,303),"",7)

7 STREET ADDRESS FREE TEXT K:\$L(X)>64!(\$L(X)<1) X

M/FORM FID # 19

28 STORED: \$p("%IC(DA,1),"",1)

44 POST ID: s %cm K:\$L(X)>64!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

8 CITY FREE TEXT K:\$L(X)>23!(\$L(X)<1)!'(XZ.AP) X

M/FORM FID # 20

28 STORED: \$p("%IC(DA,1),"",2)

44 POST ID: s %cm K:\$L(X)>23!(\$L(X)<1)!'(XZ.AP) X s:'\$d(X) cm="" s:cm("" cm=""

9 STATE POINTER 1003' DIZ(1003,

M/FORM FID # 21

20 CONVERSION CODE: s cm=\$s(cm:"":",('\$d("DIZ(1003,cm,0))#10):cm,1:\$p("DIZ(1003,cm,0),"",1))

28 STORED: \$p("%IC(DA,1),"",3)

10.5 ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 22

28 STORED: \$p("%IC(DA,1),"",6)

44 POST ID: s %cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

9.5 OVERSEAS ADDRESS FREE TEXT K:\$L(X)>64!(\$L(X)<1) X

M/FORM FID # 23

28 STORED: \$p("%IC(DA,1),"",4)

44 POST ID: s %cm K:\$L(X)>64!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

W4 Statement of official

DATA ELEMENT DATA TYPE POINTER GLOBAL REFERENCE OR INPUT VERIFICATION

D:DAILY

M/FORM FID # 89

20 CONVERSION CODE: s cm=Sp(\$p(";H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;" ;" _cm ":" ,2) ,";" ,1) ,cm=Sp(cm, ":" ,1)
28 STORED: \$p(^ICC(DA,3) ,"" ,19)

21.45 DOI OTHER RATE BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 87

20 CONVERSION CODE: s cm=Sp(\$p(";H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;" ;" _cm ":" ,2) ,";" ,1) ,cm=Sp(cm, ":" ,1)
28 STORED: \$p(^ICC(DA,3) ,"" ,21)

229.1 WORK STOPPED BASE RATE NUMERIC FORMAT S:X("\$" X=Sp(X, "\$" ,2) K:+X'=X(X'7.NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 94

28 STORED: \$p(^ICC(DA,222) ,"" ,1)

229.3 WORK STOPPED SUBSISTENCE NUMERIC FORMAT S:X("\$" X=Sp(X, "\$" ,2) K:+X'=X(X'7.NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 96

28 STORED: \$p(^ICC(DA,222) ,"" ,3)

229.4 WORK STOPPED QUARTERS NUMERIC FORMAT S:X("\$" X=Sp(X, "\$" ,2) K:+X'=X(X'7.NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 97

28 STORED: \$p(^ICC(DA,222) ,"" ,5)

229.5 WORK STOPPED OTHER RATE NUMERIC FORMAT S:X("\$" X=Sp(X, "\$" ,2) K:+X'=X(X'7.NI". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 100

28 STORED: \$p(^ICC(DA,222) ,"" ,7)

229.2 WORK STOPPED BASE BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 93

20 CONVERSION CODE: s cm=Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;"",;"_cm_":",2)",;"",1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,222)",""",2)

229.39 WORK STOPPED SUB BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M FORM FID # 95

20 CONVERSION CODE: s cm=Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;"",;"_cm_":",2)",;"",1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,222)",""",4)

229.48 WORK STOPPED QUARTERS BA SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M FORM FID # 98

20 CONVERSION CODE: s cm=Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;"",;"_cm_":",2)",;"",1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,222)",""",5)

229.58 WORK STOPPED OTHER BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M FORM FID # 95

20 CONVERSION CODE: s cm=Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;"",;"_cm_":",2)",;"",1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,222)",""",8)

30 PREMIUM PAY NUMERIC FORMAT S:X("\$" X=SP(X,"\$",2) R:X'-X&(X'~ N1'."2N)!(X>999999.99)!(X<0) X

M FORM FID # 106

28 STORED: Sp("ICC(DA,12)",""",1)

31 PREMIUM PAY BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-WEEKLY
A:ANNUALLY

D:DAILY

M/FORM FID # 114

```
20 CONVERSION CODE: s=cm$P($P("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;  
A:ANNUALLY;D:DAILY;",",",1),",",1),cm$P(cm,"",1)  
28 STORED: $P("IC(DA,12)",",",2)  
29 PREFIELD CODE: S:'SD(cbi(106)) skip="D"
```

32 NIGHT EAY NL'ERIC FORNAT S:X["\$" X=SP(X,"\$",2) K:+K'-VL(X'?.NL". "2N)!(X>999999.99)!(X<0) X

M/FORM FID # 107

28 STORED: \$p(^JOC(DA,12),"^",3)

33 NIGHT PAY BASIS	SET OF CODES	H:HOURLY W:WEEKLY M:MONTHLY B:BI-WEEKLY A:ANNUALLY D:DAILY
--------------------	--------------	---

M/FORM FID # 108

```
20 CONVERSION CODE: s=cm-$p($p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;  
A:ANNUALLY;D:DAILY;"",")_cm_"":2),";",1),cm-$p(cm,"":1)  
28 STORED: $p(^IC(DA,12),"",4)  
29 PREFIELD CODE: s:'$d(cbi(107)) skip="D"
```

34 SUNDAY FAY NUMERIC FORMAT S:X"\$" X=SP(X,"\$",2) K:+X'-X(X'?NI".2N)!(X>999999.99)!(X<0) Y

M/FORM FID # 109

28 STORED: SC(^IC(DA, 12), " ^", 5)

35 SUNDAY PAY BASIS	SET OF CODES	H:HOURLY W:WEEKLY M:MONTHLY B:BI-WEEKLY A:ANNUALLY D:DAILY
---------------------	--------------	---

M/FORM FID # 110

```
20 CONVERSION CODE: S=CM$(SP("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BIL+PERLY;  
A:ANNUALLY;D:DAILY;"");"_CM_"$:"2"),";",1),CM$(CM,";",1)  
28 STORED: SP("TCC(DA,12),"",6)  
29 PREFIELD CODE: S:'SD(abi(109)) skip="D"
```

```
36 OTHER PAY      NUMERIC FORMAT  S:X["$" X-SP(X,"$",2) K:+X'-X(X'?.N1"."2N)!(X>999999.99)!(X<0) X
```

M/FORM FTD # 111

28 STORED: Sp(IOC(DA,12),"" ,7)

37 OTHER PAY BASIS

SET OF CODES

H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-WEEKLY
A:ANNUALLY
D:DAILY

M/FORM FID # 112

20 CONVERSION CODE: s cm=Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;
A:ANNUALLY;D:DAILY;"",";",_cm_=":",2),";",1),cm=Sp(cm,"",1)
28 STORED: Sp(^ICC(DA,12),"",8)
29 PREFIELD CODE: S:'\$D(cbl(111)) skip="D"

38 OTHER PAY SPECIFY

FREE TEXT

K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 113

28 STORED: Sp(^ICC(DA,12),"",9)
29 PREFIELD CODE: S:'\$D(cbl(111)) skip="D"
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

303 REGULAR WORK SCHEDULE

FREE TEXT

K:\$L(X)>15!(\$L(X)<3) X

M/FORM FID # 24

28 STORED: Sp(^ICC(DA,300),"",4)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<3) X s:'\$d(X) cm="" s:cm(" " cm=""

605 EMPLOYED IN DUTY 11 MOS

SET OF CODES

Y:YES
N:NO

M/FORM FID # 25

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO;"",";",_cm_=":",2),";",1),cm=Sp(cm,"",1)
28 STORED: Sp(^ICC(DA,600),"",2)

606 DUTY AFFORDED 11 MOS WOR

SET OF CODES

Y:YES
N:NO

M/FORM FID # 26

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO;"",";",_cm_=":",2),";",1),cm=Sp(cm,"",1)
28 STORED: Sp(^ICC(DA,600),"",3)
29 PREFIELD CODE:
I '\$D(cbl(25)) s skip="D",cbl(0,26)=""
i \$d(cbl(25)) s:cbl(25)["Y"!cbl(25)["Y"] skip="D",cbl(0,26)=""

727 TOTAL YRS CIV SERVICE

NUMERIC FORMAT

K:+X'=X1(X>50)!(X<1)!(X?.E1". "IN.N) X

M/FORM FID # 47

28 STORED: Sp(^ICC(DA,704),"",2)

727.5 TOTAL MOS CIV SERVICE NUMERIC FORMAT K:+X=X!(X>11)!(X<1)!(X7.E1)".IN.N) X

M/FORM FID # 48

28 STORED: \$p("ICC(DA,700),"",5)

W5 HEALTH/LEAVE/CDP

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
728 EMP ENROLLED IN FEHB	SET OF CODES	Y:YES N:NO

M/FORM FID # 28

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",";","_cm_":",2)",";","1),cm=\$p(cm,";",1)
28 STORED: \$p("ICC(DA,704),"",3)

25 FEHB CODE FREE TEXT K:\$L(X)>15!(SL(X)<1) X

M/FORM FID # 30

28 STORED: \$p("ICC(DA,3),"",13)
29 PREFIELD CODE:

I '\$D(chi(28)) s skip="D"
i \$J(chi(28)) s:chi(28)!"N"!(chi(28)!"n") skip="D"

44 POST ID: s X=cm K:\$L(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm"" cm=""

728.1 PAY PERIOD END DATE FOR DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 44

20 CONVERSION CODE:
s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)"=" X=X " at " \$e(cm"000",9,10)" " \$e(cm"000",11,12) s X=X "
s:X="(1700)" X="" s cm=X

28 STORED: \$p("ICC(DA,700),"",6)

729 EMP ENROLLED IN OPT LIFE SET OF CODES Y:YES
N:NO

M/FORM FID # 29

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",";","_cm_":",2)",";","1),cm=\$p(cm,";",1)
28 STORED: \$p("ICC(DA,704),"",4)

729.1 LIFE INS OPTION SET OF CODES A:A
B:B
C:C

M/FORM FID # 83

20 CONVERSION CODE: s cm=Sp(Sp("A:A;B:B;C:C;",";"_cm_":",2),";",1),cm=Sp(cm,";",1)
 28 STORED: Sp(^ICC(DA,704),"",5)

729.2 HOW MANY OPT/B MULTIPLES FREE TEXT K:\$L(X)>30!(\$L(X)<3) X

M/FORM FID # 32

28 STORED: Sp(^ICC(DA,704),"",6)
 29 PREFIELD CODE:
 I 'SD(cbi(31)) s skip="D"
 i \$d(cbi(31)) s:cbi(31){"B" skip="D"

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<3) X s:'\$d(X) cm="" s:cm[""] cm=""

252 SICK LEAVE FROM DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 115

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",3)

252.5 SICK LEAVE THRU DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 116

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",4)

29 PREFIELD CODE: s:'\$d(cbi(115)) skip="D"

253 ANNUAL LEAVE FROM DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 117

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",5)

253.5 ANNUAL LEAVE THRU DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 118

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X_"")
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,250),"",6)

29 PREFIELD CODE: S:'\$D(cbi(117)) skip="D"

251 OTHER LEAVE FROM DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 119

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X_"")
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,250),"",7)

251.5 OTHER LEAVE THRU DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 120

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X_"")
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,250),"",8)

29 PREFIELD CODE: s:'\$d(cbi(119)) skip="D"

251.6 OTHER LEAVE SPECIFY FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 121

28 STORED: \$p(^ICC(DA,250),"",9)

29 PREFIELD CODE: S:'\$d(cbi(119)) skip="D"

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

250 COP RECVD FROM DATE S %DT="E" D ^%DT S X=Y K:Y<1 X

M/FORM FID # 123

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s X=X_"")
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,250),"",1)

5 COP RECVD THRU DATE S %DT="E" D ^%DT S %Y K:Y<1 X

M/FORM FID # 122

20 CONVERSION CODE:

s %="(" s:+\$e(cm,4,5) %X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %X \$e(cm,6,7) ", " s %X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" %X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s %X "
s:%="(1700)" %=" s cmX

28 STORED: \$p(^ICC(DA,250),"",2)

29 PREFIELD CODE: S:'\$D(cbi(123)) skip="D"

230 D/T BAY STOPPED DATE S %DT="E" D ^%DT S %Y K:Y<1 X

M/FORM FID # 33

20 CONVERSION CODE:

s %="(" s:+\$e(cm,4,5) %X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %X \$e(cm,6,7) ", " s %X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" %X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s %X "
s:%="(1700)" %=" s cmX

28 STORED: \$p(^ICC(DA,221),"",2)

M/FORM FID # 49

20 CONVERSION CODE:

28 STORED: nowhere

44 POST ID:

M/FORM FID # 50

20 CONVERSION CODE:

28 STORED: nowhere

44 POST ID:

WS CERTIFICATION

DATA ELEMENT DATA TYPE POINTER GLOBAL REFERENCE OR INPUT VERIFICATION

231 D/T RETURNED TO WORK DATE S %DT="E" D ^%DT S %Y K:Y<1 X

M/FORM FID # 34

20 CONVERSION CODE:

s %="(" s:+\$e(cm,4,5) %X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) %X \$e(cm,6,7) ", " s %X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" %X " at " \$e(cm"000",9,10) ":" \$e(cm"000",11,12) s %X "
s:%="(1700)" %=" s cmX

28 STORED: \$p(^ICC(DA,221),"",3)

324 NEW WORK WEEK

FREE TEXT

K:\$L(X)>30!(\$L(X)<3) X

M/FORM FID # 43

28 STORED: \$p(^ICC(DA,303),"",8)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<3) X s:\$d(X) cm="" s:cm "" cm=""

311 DISABLED FOR WORK

SET OF CODES

Y:YES

N:NO

M/FORM FID # 35

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"","_cm_":",2),"",1),cm=\$p(cm,"",1)

28 STORED: \$p(^ICC(DA,303),"",2)

320 DESCRIBE NEW WORK ASSIGN

M/FORM FID # 36

28 STORED: nowhere

29 PREFIELD CODE:

I '\$D(cbi(35)) s skip="D"

i \$d(cbi(35)) s:cbi(35){"n":(cbi(35){"N"} skip="D"

s:\$d(lvl) lvl=0 i \$d(cbi(0))>9 s s:file=1 d filecm~ptrfile1 k shhfile

i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)

+d upcntr~fmfm s New=0,cbi(0)="" k file

s lvl=lvl+1

i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)

s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

16 CALLS M/FORM: DESCRIBE NEW DUTIES (#630)

M/FORM: DESCRIBE NEW DUTIES (#630)

I. FHS-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$H),DT="" d ~DT s DT=Y k X,Y
d sec~fmfm
d crink~fmfm

II. WINDOWS

DESCRIBE NEW DUTIES Window 1

DATA ELEMENT

DATA TYPE

POINTER GLOBAL REFERENCE OR INPUT VERIFICATION

.01 DESCRIBE NEW WORK ASSING

WORD PROCESSING

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),311,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$e(\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

231.1 RETURNED TO WORK BASE RA NUMERIC FORMAT S:X("\$" X=\$P(X,"\$",2) K:~X'=X\$(X'?.NL"."2N)!(X>999999.99)!(X<0) X

M/FORM FID # 125

28 STORED: \$p(^ICC(DA,231),"",1)

231.2 RETURNED TO WORK BASE BA SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-MONTHLY
A:ANNUALLY
D:DAILY

M/FORM FID # 126

20 CONVERSION CODE: s cm=\$p(\$p("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-MONTHLY;
A:ANNUALLY;D:DAILY;"",",",_cm_",2),",",1),cm=\$p(cm,"",1)

28 STORED: \$p(^ICC(DA,231),"",2)

29 PREFIELD CODE: s:\$d(cbi(125)) skip="D"

720 OFFICIAL REMARKS FREE TEXT K:SL(X)>50!(SL(X)<1) X

M/FORM FID # 51

28 STORED: \$p(^ICC(DA,402),"",6)

44 POST ID: s X=cm K:SL(X)>50!(SL(X)<1) X s:\$d(X) cm="" s:cm{""" cm=""

402 SUPERVISOR'S NAME FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 55

28 STORED: \$p(^ICC(DA,402),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:\$d(X) cm="" s:cm{""" cm=""

402.4 DATE SUPERVISOR SIGNED DATE S ^DT="E" D ^TOT S X=Y K:Y<1 X

M/FORM FID # 79

20 CONVERSION CODE:

s X=\$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",+\$e(cm,4,5))_ " s:+\$e(cm,6,7) X=X_ \$e(cm,6,7)_ " s
X=X_(1700+\$e(cm,1,3))

s:\$p(cm,"",2)'="" X=X_ at " _\$e(cm "000",9,10)_" :_" _\$e(cm "000",11,12) s X=X_ "
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,402),"",5)

402.1 SUPERVISOR'S TITLE FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 56

28 STORED: Sp(ICC(DA,402),"",2)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

14 REPORTING OFFICE POINIER 1063' INJ(1063,

M/FORM FID # 58

20 CONVERSION CODE: s cm=\$s(cm="" ":" , '\$d(INJ(1063,cm,0))#10):cm,1:Sp(INJ(1063,cm,0),"",1))

28 STORED: Sp(ICC(DA,10),"",1)

14.01 RO ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 59

28 STORED: Sp(ICC(DA,10),"",2)

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

14.02 RO CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 61

28 STORED: Sp(ICC(DA,10),"",3)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

14.03 RO STATE POINIER 1003' DIZ(1003,

M/FORM FID # 60

20 CONVERSION CODE: s cm=\$s(cm="" ":" , '\$d(DIZ(1003,cm,0))#10):cm,1:Sp(DIZ(1003,cm,0),"",1))

28 STORED: Sp(ICC(DA,10),"",4)

14.04 RO ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 62

28 STORED: Sp(ICC(DA,10),"",5)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

731 PERSON TO CONTACT SET OF CODES S: SUPERVISOR
O: OTHER

M/FORM FID # 81

20 CONVERSION CODE: s cm=Sp(Sp(" ; S: SUPERVISOR ; O: OTHER ; " ; " _cm _ " : " , 2) ; " , 1) , cm=Sp(cm , " : " , 1)

28 STORED: Sp(ICC(DA,700),"",1)

14.05 POINT OF CONTACT FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 80

28 STORED: \$p(^ICC(DA,10),"",6)

44 POST ID: s X=am K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) am="" s:am["" am=""

14.06 FCC PHONE

FREE TEXT

K:\$L(X)>13!(\$L(X)<10) X

M/FORM FID # 54

28 STORED: \$p(^ICC(DA,10),"",7)

44 POST ID: s X=am K:\$L(X)>13!(\$L(X)<10) X s:'\$d(X) am="" s:am["" am=""

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fmfm
i file=1!(\$d(ERFLAG)) d OA7^T3CPNF,^T3CPLE

CA-8 CLAIM FOR CONTINUING COMPENSATION ON ACCOUNT OF DISABILITY
STATEMENT OF INJURED EMPLOYEE

1. NAME OF INJURED EMPLOYEE _____
2. OWCP FILE NUMBER, IF KNOWN _____
3. HOME MAILING ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
OVERSEAS ADDRESS _____
4. SOCIAL SECURITY NUMBER _____
5. DATE AND HOUR OF INJURY _____
6. PERIOD COMPENSATION CLAIMED, FROM _____ THROUGH _____
7. HAVE YOU RECEIVED ANY LEAVE PAY? _____
SICK LEAVE FROM _____ THROUGH _____
ANNUAL LEAVE FROM _____ THROUGH _____
OTHER LEAVE FROM _____ THROUGH _____
SPECIFY OTHER LEAVE _____
8. DO YOU WISH TO REPURCHASE LEAVE? _____
9. COMPLETE (A & B) IF YOU WORKED DURING THE PERIOD SHOWN IN ITEM #6.
A. SALARIED EMPLOYMENT Press <Return> for WORK INFORMATION
B. Press <Return> for COMMISSION & SELF-EMPLOYMENT
10. IF YOU HAVE APPLIED FOR EMPLOYMENT WITH THE U.S. TRAINING AND
EMPLOYMENT SERVICE GIVE THE FOLLOWING:
REGISTRATION NO. _____
DATE OF REGISTRATION _____
OFFICE ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
11. *Press <Return> for STATE REASON FOR NO WORK*
12. IF, SINCE FILING YOUR INITIAL CLAIM FOR COMPENSATION, YOU HAVE
APPLIED FOR OR RECEIVED VA BENEFITS BASE ON MILITARY SERVICE FOR
THE U.S., GIVE:
CLAIM NO. _____
NATURE OF DISABILITY _____
MONTHLY PAYMENT \$ _____
NAME OFFICE WHERE CLAIM IS FILED _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
13. IF, SINCE FILING YOUR INITIAL CLAIM FOR COMPENSATION, YOU HAVE
APPLIED FOR OR RECEIVED ANNUITY UNDER CIVIL SERVICE RETIREMENT
ACT, GIVE:
CLAIM NO. _____
AMOUNT OF MONTHLY PAYMENT \$ _____
NAME OF OFFICE WHERE CLAIM IS FILED _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
15. DATE EMPLOYEE SIGNED (MO., DAY, YEAR) _____

CA-8 CON'T

STATEMENT OF OFFICIAL SUPERIOR

16. EMPLOYEE HAS RETURNED TO WORK, SHOW DATE _____
17. EMP'S WORK WEEK ON RETURN TO DUTY _____
18. HAS EMP RECEIVED ANY PAY FROM AGENCY? _____
19. IF ANSWER TO #18 IS YES, SHOW:
AMOUNT: \$ _____
TYPE OF PAYMENT: _____
PERIOD: FROM: _____ THROUGH: _____
PAYMENT RATE: \$ _____ PAYMENT BASIS: _____
0. *Press <Return> for EXPLAIN CHGS IN EMP HLTH*
23. TITLE _____
24. DATE (MO., DAY, YEAR) _____

WORK INFORMATION

DATES & HOURS WORKED _____
PAY RATE _____ TOTAL AMOUNT EARNED _____
TYPE WORK PERFORMED _____
EMPLOYER NAME _____
EMPLOYER STREET ADDRESS _____
EMPLOYER CITY/STATE/ZIP CODE _____

COMMISSION AND SELF-EMPLOYMENT

DATES & HOURS WORKED _____
NAME OF BUSINESS _____
STREET ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
TYPE OF ACTIVITY PERFORMED _____
INCOME DERIVED _____

*Word Processing Window will appear.

I. PRE-FORM CODE

```

n filen,rptfile,filing s:'$d(nocomp) filen=^ptc(iform,99)
+s:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1,indexist=2 k ident
s U="" i '$d(DT) s X=$zd($H),%DT="" d ^DT s DT=Y k X,Y
S:$X(DA) $P(^ICC(DA,221),U,2,3)="" d sec^fntm
n filen,rptfile,filing s:'$d(nocomp) filen=^ptc(iform,99)

```

II. WINDOWS

WI employee statement

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	POINTER 1004	EMPLOY(
--------	--------------	---------

M/FORM FID # 62

```

17 EXTERNAL DEFAULT: NAME
18 INTERNAL DEFAULT: NAMEPTR
20 CONVERSION CODE: s cm=$s(cm:"":",",('$d(^EMPLOY(cm,0))#10):cm,1:$p(^EMPLOY(cm,0),"",1))
28 STORED: $p(^ICC(DA,0),"",3)
32 POST FILING CODE: d setindx^fntm
44 POST ID: i $d(new) s:'$d(^ICC("C",new,x1)) msg="Claim not logged",y=24,cm=""
54 INDEX: ^ICC("C")
55 LOOKUP NODE:
    62^40^$p(^ICC(Lids,0),"",3)
    9^23^$p(^ICC(Lids,0),"",2)

```

137 OACP FILE NUMBER	FREE TEXT	K:SL(X)>13!:(SL(X)<1) X
----------------------	-----------	-------------------------

M/FORM FID # 2

```

28 STORED: $p(^ICC(DA,130),"",6)
44 POST ID: s X=cm K:SL(X)>13!:(SL(X)<1) X s:'$d(X) cm="" s:cm[""] cm=""

```

7 STREET ADDRESS	FREE TEXT	K:SL(X)>64!:(SL(X)<1) X
------------------	-----------	-------------------------

M/FORM FID # 3

```

28 STORED: $p(^ICC(DA,1),"",1)
44 POST ID: s X=cm K:SL(X)>64!:(SL(X)<1) X s:'$d(X) cm="" s:cm[""] cm=""

```

8 CITY	FREE TEXT	K:SL(X)>23!:(SL(X)<1)!^(X2.AP) X
--------	-----------	----------------------------------

M/FORM FID # 4

```

28 STORED: $p(^ICC(DA,1),"",2)
44 POST ID: s X=cm K:SL(X)>23!:(SL(X)<1)!^(X2.AP) X s:'$d(X) cm="" s:cm[""] cm=""

```

9 STATE

POINTER 1003' DIZ(1003,

M/FORM FID # 5

20 CONVERSION CODE: s cm=6s(cm="": "", '(\$d(^DIZ(1003,cm,0))#10):cm.1:Sp(^DIZ(1003,cm,0),"",1))
28 STORED: Sp(^IOC(DA,1),"",3)

10.5 ZIP CODE

FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 6

28 STORED: Sp(^IOC(DA,1),"",6)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm(" " cm=""

9.5 OVERSEAS ADDRESS

FREE TEXT K:SL(X)>64!(SL(X)<1) X

M/FORM FID # 7

28 STORED: Sp(^IOC(DA,1),"",4)
44 POST ID: s X=cm K:SL(X)>64!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

3 SSN

FREE TEXT K:SL(X)>9!(SL(X)<9)!('X79N) X

M/FORM FID # 8

28 STORED: Sp(^IOC(DA,0),"",4)
44 POST ID: s X=cm K:SL(X)>9!(SL(X)<9)!('X79N) X s:'\$d(X) cm="" s:cm(" " cm=""

1 INJURY/ILLNESS DATE & TI

DATE

S ^DT="E" D ^DT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 9

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) " : " Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^IOC(DA,0),"",2)
32 POST FILING CODE: d setindx^fmcm
54 INDEX: ^IOC("D")
55 LOOKUP NODE:
9^23^Sp(^IOC(Lids,0),"",2)
62^40^Sp(^IOC(Lids,0),"",3)

701 DATE WAGES LOST FROM

DATE

S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 37

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) " : " Se(cm "000",11,12) s X=X ")"

s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,700), "", 7)

44 POST ID: s X=cm k:X<Sp(chi(9),db,2) X i 'Sd(X) s cm="" s:cm("" cm=""

701.5 DATES WAGES LOST THRU DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 38

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ". ",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " , " s X=X (1700+Se(cm,1,3))
s:Sp(cm, ". ", 2) '="" X=X " at " Se(cm "000", 9,10) " : " Se(cm "000", 11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,700), "", 9)

44 POST ID:

i 'Sd(chi(37)) s skip="D"

s X=cm k:X<Sp(chi(37),db,2) X i 'Sd(X) s cm="" s:cm("" cm=""

806 PAY REC'D DURING COMP SET OF CODES Y:YES
N:NO

M/FORM FID # 36

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO; ", " " cm ":", 2), ":", 1), cm=Sp(cm, ":", 1)

28 STORED: Sp(^ICC(DA,704), "", 7)

252 SICK LEAVE FROM DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 63

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ". ",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " , " s X=X (1700+Se(cm,1,3))
s:Sp(cm, ". ", 2) '="" X=X " at " Se(cm "000", 9,10) " : " Se(cm "000", 11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250), "", 3)

29 PREFIELD CODE: S:'Sd(chi(36)) skip="D" i 'Sd(skip) s:chi(36)' ("Y" skip="D"

252.5 SICK LEAVE THRU DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 64

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ". ",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " , " s X=X (1700+Se(cm,1,3))
s:Sp(cm, ". ", 2) '="" X=X " at " Se(cm "000", 9,10) " : " Se(cm "000", 11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250), "", 4)

29 PREFIELD CODE:

S:'\$D(cbi(36)) skip="D" i '\$d(skip) s:cbi(36)'("Y" skip="D"
s:'\$d(cbi(63)) skip="D"

253 ANNUAL LEAVE FROM DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 65

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X "
s:X="(1700)" X="" s cmX

28 STORED: \$p(^IOC(DA,250),"",5)

29 PREFIELD CODE: s:'\$d(cbi(36)) skip="D" i '\$d(skip) s:cbi(36)'("Y" skip="D"

253.5 ANNUAL LEAVE THRU DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 66

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X "
s:X="(1700)" X="" s cmX

28 STORED: \$p(^IOC(DA,250),"",6)

29 PREFIELD CODE:

S:'\$D(cbi(36)) skip="D" i '\$d(skip) s:cbi(36)'("Y" skip="D"
s:'\$d(cbi(65)) skip="D"

251 OTHER LEAVE FROM DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 67

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X "
s:X="(1700)" X="" s cmX

28 STORED: \$p(^IOC(DA,250),"",7)

29 PREFIELD CODE: s:'\$d(cbi(36)) skip="D" i '\$d(skip) s:cbi(36)'("Y" skip="D"

251.5 OTHER LEAVE THRU DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 68

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at "\$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X "
s:X="(1700)" X="" s cmX

s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,250),"",8)

29 PREFIELD CODE: S:'\$D(cbi(67)) skip="D"

251.6 OTHER LEAVE SPECIFY FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 69

28 STORED: Sp(^ICC(DA,250),"",9)

29 PREFIELD CODE: S:'\$D(cbi(67)) skip="D"

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

801 REPURCHASE LEAVE SET OF CODES Y:YES
N:NO

M/FORM FID # 10

20 CONVERSION CODE: s cm=Sp(\$p("Y:YES;N:NO","";"_cm_":",2",";"),1),cm=Sp(cm,"",1)

28 STORED: Sp(^ICC(DA,704),"",1)

2217 WORK INFO DURING COP/COM

M/FORM FID # 11

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d fileone~ptrfile1 k shhfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fmr s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0-DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

16 CALLS M/FORM: WORK INFORMATION (#655)

M/FORM: WORK INFORMATION (#655)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$ad(\$H),MDT="" d ^MDT s DT=Y k X,Y
d sec~fmr

II. WINDOWS

WORK INFORMATION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DATES & HOURS WORKED	FREE TEXT	K:X[""]!(\$A(X)=45) X I \$D(X) K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 1		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",1)		
32 POST FILING CODE: d setindk^fmfm		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""		
54 INDEX: ^ICC(DA(1),212,"B")		
55 LOOKUP NODE: 1^30^\$p(^ICC(DA(1),212,lids,0),"",1)		
1 PAY RATE	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 2		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",2)		
44 POST ID: s X= a K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""		
2 TOTAL AMOUNT EARNED	NUMERIC FORMAT	S:X["\$"] X=\$P(X,"\$",2) K:+X'-X\$(X'7.N1"."2N)!(X>99999)!(X<0) X
M/FORM FID # 3		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",3)		
3 TYPE WORK PERFORMED	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
M/FORM FID # 4		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",4)		
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""		
4 EMPLOYER NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 5		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",5)		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""		
5 EMPLOYER STREET ADDRESS	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 6		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",6)		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""		
6 EMPLOYER CITY/STATE/ZIP	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 7		
28 STORED: \$p(^ICC(DA(1),212,DA,0),"",7)		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""		

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0`fmfm
```

808 COMMISSION & SELF-EMP

M/FORM FID # 12

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone`ptrfile1 k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),Dl s lvl=lvl-1
```

16 CALLS M/FORM: COMMISSION & SELF-EMPLOYMENT (#665)

M/FORM: COMMISSION & SELF-EMPLOYMENT (#665)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=`ptc(iform,99)
+s:$d(nocomp) filen=`ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),TOT="" d `TOT s DT=Y k X,Y
d sec`fmfm
```

II. WINDOWS

COMMISSION & SELF-EMPLOYMENT Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DATES & HOURS WORKED	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 1

```
28 STORED: $p(`ICC(DA(1),812,DA,0),"",1)
32 POST FILING CODE: d setinrk`fmfm
44 POST ID: s X=cm K:$L(X)>20!($L(X)<1) X s:'$d(X) cm="" s:cm(" " cm=""
54 INDEX: `ICC(DA(1),812,"B")
55 LOOKUP NODE: 1`20`$p(`ICC(DA(1),812,lids,0),"",1)
```

1 NAME OF BUSINESS	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
--------------------	-----------	--------------------------

M/FORM FID # 6

28 STORED: \$p(^ICC(DA(1),812,DA,0),"",2)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

1.1 STREET ADDRESS FREE TEXT K:\$L(X)>45!(\$L(X)<1) X

M/FORM FID # 7

28 STORED: \$p(^ICC(DA(1),812,DA,0),"",6)
44 POST ID: s X=cm K:\$L(X)>45!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

1.2 CITY FREE TEXT K:\$L(X)>23!(\$L(X)<1) X

M/FORM FID # 8

28 STORED: \$p(^ICC(DA(1),812,DA,0),"",7)
44 POST ID: s X=cm K:\$L(X)>23!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

1.3 STATE POINTER 1003' DIZ(1003,

M/FORM FID # 13

20 CONVERSION CODE: s cm=\$s(cm:"":'(\$d(DIZ(1003,cm,0))#10):cm,1:\$p(DIZ(1003,cm,0),"",1))
28 STORED: \$p(^ICC(DA(1),812,DA,0),"",8)

1.4 ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 10

28 STORED: \$p(^ICC(DA(1),812,DA,0),"",9)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

3 TYPE OF ACTIVITY PERFORM FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 11

28 STORED: \$p(^ICC(DA(1),812,DA,0),"",4)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

4 INCOME DERIVED FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 5

28 STORED: \$p(^ICC(DA(1),812,DA,0),"",5)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0^filing

W2 TRAINING

DATA ELEMENT	DATA TYPE	FOINIER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

811 TRAINING REGISTRATION NU	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
------------------------------	-----------	--------------------------

M/FORM FID # 39

28 STORED: \$p(^ICC(DA,804),"",1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

811.5 TRAINING DATE OF REGISTR	DATE	S %DT="E" D %DT S X=Y K:Y<1 X
--------------------------------	------	-------------------------------

M/FORM FID # 40

20 CONVERSION CODE:

s X=" " s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " \$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X "
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,804),"",6)

811.1 TRAINING STREET ADDRESS	FREE TEXT	K:\$L(X)>54!(\$L(X)<1) X
-------------------------------	-----------	--------------------------

M/FORM FID # 41

28 STORED: \$p(^ICC(DA,804),"",2)

44 POST ID: s X=cm K:\$L(X)>54!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

811.2 TRAINING CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
---------------------	-----------	--------------------------

M/FORM FID # 42

28 STORED: \$p(^ICC(DA,804),"",3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

811.3 TRAINING STATE	FOINIER 1003' DIZ(1003,
----------------------	-------------------------

M/FORM FID # 43

20 CONVERSION CODE: s cm=\$e(cm:"":'(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,804),"",4)

811.4 TRAINING ZIP CODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
-------------------------	-----------	--------------------------

M/FORM FID # 44

28 STORED: \$p(^ICC(DA,804),"",5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm[""] cm=""

802 STATE REASON FOR NOT WOR

M/FORM FID # 16

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone^ptrfilel k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,l:file)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: STATE REASON FOR NO WORK (#666)

M/FORM: STATE REASON FOR NO WORK (#666)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=^ptc(iform,99)
+s:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=^d(SH),DT="" d ^DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm
```

II. WINDOWS

STATE REASON FOR NO WORK Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 STATE REASON FOR NOT WOR	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),801,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,l:file)
+d shuffle^fmfm d node0^fmfm
```

710 VA CLAIM NUMBER FREE TEXT K:\$L(X)>15!(\$L(X)<1) X

M/FORM FID # 45

28 STORED: \$p(^ICC(DA,701),"",1)

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

710.6 VA NATURE OF DISABILITY FREE TEXT K:SL(X)>57!(SL(X)<1) X

M/FORM FID # 46

28 STORED: Sp('ICC(DA,701),""",7)

44 POST ID: s X=cm K:SL(X)>57!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

710.7 VA AMOUNT OF PAYMENT NUMERIC FORMAT S:X"\$" X=SP(X,"\$",2) K:+X'=X(X'?.N1"."2N)!(X>99999.99)!(X<0) X

M/FORM FID # 47

28 STORED: Sp('ICC(DA,701),""",8)

710.1 VA OFFICE WHERE CLAIM FI FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 48

28 STORED: Sp('ICC(DA,701),""",2)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

710.2 VA STREET ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 49

28 STORED: Sp('ICC(DA,701),""",3)

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

710.3 VA CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 50

28 STORED: Sp('ICC(DA,701),""",4)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm[""" cm=""

710.4 VA STATE POINTER 1003' DIZ(1003,

M/FORM FID # 51

20 CONVERSION CODE: s cm=\$s(cm=""",',(\$d('DIZ(1003,cm,0))#10):cm,1:Sp('DIZ(1003,cm,0),""",1))
28 STORED: Sp('ICC(DA,701),""",5)

710.5 VA ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 52

28 STORED: Sp('ICC(DA,701),""",6)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm[""" cm=""

WS Civil Service Activity

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

714 OTHER ANNUITY'S CLAIM NU FREE TEXT K:SL(X)>15!(SL(X)<1) X

M/FORM FID # 53

28 STORED: Sp(^ICC(DA,707),"",1)

44 POST ID: s X=cm K:SL(X)>15!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

714.2 OTHER ANNUITY AMOUNT NUMERIC FORMAT S:X("\$" X=\$P(X,"\$",2) K:~X~X(X'7.NI".2N)!(X>9999999.99)!X<0) X

M/FORM FID # 54

28 STORED: Sp(^ICC(DA,707),"",3)

714.3 OTHER ANNUITY OFFICE FREE TEXT K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 55

28 STORED: Sp(^ICC(DA,707),"",4)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

714.4 OTHER ANNUITY STREET ADD FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 56

28 STORED: Sp(^ICC(DA,707),"",5)

44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

714.5 OTHER ANNUITY CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 57

28 STORED: Sp(^ICC(DA,707),"",6)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

714.6 OTHER ANNUITY STATE FOUNDER 1003' DIZ(1003,

M/FORM FID # 58

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: Sp(^ICC(DA,707),"",7)

714.7 OTHER ANNUITY ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 59

28 STORED: Sp(^ICC(DA,707),"",8)

44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

322 DATE EMP/CLAIMANT SIGNED DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 24

20 CONVERSION CODE:

s X="(" s:+\$s(cm,4,5) X=X_Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",

+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,303),"",7)

WM OFFICIAL SUPERIOR

DATA ELEMENT	DATA TYPE	FOINDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

231 D/T RETURNED TO WORK	DATE	S %DT="ET" D %DT S X=Y K:Y<1 X
--------------------------	------	--------------------------------

M/FORM FID # 25

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,221),"",3)

324 NEW WORK WEEK	FREE TEXT	K:SL(X)>30!(SL(X)<3) X
-------------------	-----------	------------------------

M/FORM FID # 26

28 STORED: Sp("ICC(DA,303),"",8)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<3) X s:'\$d(X) cm="" s:cm(" " cm=""

812 EMP RECEIVED MONIES FROM	SET OF CODES	Y:YES N:NO
------------------------------	--------------	---------------

M/FORM FID # 27

20 CONVERSION CODE: s cm=Sp(\$p("Y:YES/N:NO",";","_cm ":"",2),"",1),cm=Sp(cm,"",1)

28 STORED: Sp("ICC(DA,704),"",8)

257.2 MISC PAY RECVD GROSS	NUMERIC FORMAT	S:X "\$" X=Sp(X,"\$",2) K:+X'=X(X'?.N1". "2N)!(X0999999.99)!(X<0) X
----------------------------	----------------	---

M/FORM FID # 71

28 STORED: Sp("ICC(DA,252),"",3)

29 PREFIELD CODE: S:'\$d(chi(27)) skip="D" i '\$d(skip) s:chi(27)'{"Y" skip="D"

257.5 MISC SPECIFY	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
--------------------	-----------	------------------------

M/FORM FID # 75

28 STORED: Sp("ICC(DA,252),"",6)

29 PREFIELD CODE: s:'\$d(chi(27)) skip="D" i '\$d(skip) s:chi(27)'{"Y" skip="D"

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

257 MISC PAY RECVD FROM	DATE	S %DT="E" D %DT S X=Y K:Y<1 X
-------------------------	------	-------------------------------

M/FORM FID # 70

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)="-" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,252),"",1)

29 PREFIELD CODE: S:'\$D(cbi(27)) skip="D" i '\$d(skip) s:cbi(27)'("Y" skip="D"

257.1 MISC PAY RECVD THRU DATE S WDT="E" D ^WDT S X=Y K:Ycl X

M/FORM FID # 73

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)="-" X=X " at " Se(cm "000",9,10) ":" Se(cm "000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,252),"",2)

29 PREFIELD CODE: S:'\$D(cbi(27)) skip="D" i '\$d(skip) s:cbi(27)'("Y" skip="D"

257.3 MISC RATE NUMERIC FORMAT S:X "\$" X=SP(X,"\$",2) K:+X=X(X^7.NI"."2N)1(X>999999.99)1(X<0) X

M/FORM FID # 72

28 STORED: Sp(^ICC(DA,252),"",4)

29 PREFIELD CODE: S:'\$D(cbi(27)) skip="D" i '\$d(skip) s:cbi(27)'("Y" skip="D"

257.4 MISC RATE BASIS SET OF CODES H:HOURLY
W:WEEKLY
M:MONTHLY
B:BI-WEEKLY
A:ANNUALLY
D:DAILY

M/FORM FID # 74

20 CONVERSION CODE: s cm=Sp(Sp("H:HOURLY;W:WEEKLY;M:MONTHLY;B:BI-WEEKLY;A:ANNUALLY;
D:DAILY";",") cm ":",2),";",1),cm=Sp(cm,".",1)

28 STORED: Sp(^ICC(DA,252),"",5)

29 PREFIELD CODE: S:'\$D(cbi(27)) skip="D" i '\$d(skip) s:cbi(27)'("Y" skip="D"

816 EXPLAIN CHGS IN EMP HLEN

M/FORM FID # 30

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=l d fileone^ptrfilel k shhfile
i \$d(file),file,New=0 s filing=\$d(filing):lvl+1,lvl<filing:lvl+1,l:filing)
+d upcntr^finfm s New=0,cbi(0)="" k file

```
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
:=lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: EXPLAIN CHGS IN EMP HLTH (#670)

M/FORM: EXPLAIN CHGS IN EMP HLTH (#670)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nccomp) filen=ptc(iform,99)
+s:$d(nccomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),MDT="" d ~MDT s DT=Y k X,Y
d sec`fmfm
d crink`fmfm
```

II. WINDOWS

EXPLAIN CHGS IN EMP HLTH Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN CHGS IN EMP HLTH	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p`IOC(DA(1),805,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filling=$s('$d(filing):lvl,lvl>filling:lvl,1:filling)
+d shuffle`fmfm d node0`fmfm
```

402.1 SUPERVISOR'S TITLE	FREE TEXT	K:SL(X)>30:(SL(X)<1) X
--------------------------	-----------	------------------------

M/FORM FID # 60

28 STORED: \$p`IOC(DA,402),"",2)

44 POST ID: s X=K:SL(X)>30:(SL(X)<1) X s:\$d(X) cm="" s:cm[""] cm=""

402.4 DATE SUPERVISOR SIGNED	DATE	S MDT="E" D ~MDT S X=Y K:Y<1 X
------------------------------	------	--------------------------------

M/FORM FID # 61

20 CONVERSION CODE:


```

s X="(" s:+$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700:$e(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " $e(cm"000",9,10) ":" $e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cmX

```

28 STORED: Sp(`ICC(DA,402),""",5)

III. POST-FORM CODE

```

i $d(file),file,new=C s:'$d(lvl) lvl=0
+s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr'fmfm
i file=1!($d(ERFLAG)) d OAS`T3CPNF,T3CPFILE

```

CA-16 AUTHORIZATION FOR EXAM AND/OR TREATMENT

PART A—AUTHORIZATION

1. MEDICAL CARE FACILITY _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
2. EMPLOYEE'S NAME _____
3. DATE OF INJURY _____
4. OCCUPATION _____
5. DESCRIBE INJURY OR DISEASE: _____
Press <Return> for NATURE OF INJURY/ILLNESS
6. AUTHORIZATION OF MEDICAL CARE
A. FEES FOR SERVICES NEED PRIOR APPROVAL _____
B. TREATMENT/EXAM FOR CAUSE _____
7. IF A DISEASE OR ILLNESS IS INVOLVED,
AUTHORIZING OFFICIAL NAME _____
AUTHORIZING OFFICIAL TITLE _____
8. SIGNATURE OF AUTHORIZING OFFICIAL _____
9. TITLE OF AUTHORIZING OFFICIAL _____
10. LOCAL AGENCY TELEPHONE NUMBER _____ 11. DATE _____
12. REPORTING OFFICE _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
13. DUTY STATION _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

PART B—ATTENDING PHYSICIAN'S REPORT

15. *Press <Return> for HISTORY GIVEN BY EMPLOYEE*
16. EVIDENCE OF PRE-EXISTING CONDITION? _____
IF YES, *Press <Return> for DESCRIBE CONDITION*
17. WHAT ARE YOUR FINDINGS? _____
Press <Return> for DESCRIBE CLINICAL FINDING
18. WHAT IS YOUR DIAGNOSIS? _____
Press <Return> for DIAGNOSIS OF CONDITION
19. WAS IT CAUSED/AGGRAVATED BY EMPLOYMENT? _____
EXPLAIN ANSWER IF THERE'S DOUBT _____
Press <Return> for DOCTOR'S EXPLANATION
20. DID INJURY REQUIRE HOSPITALIZATION? _____
IF YES, ADMISSION DATE _____ DISCHARGE DATE _____
21. IS ADDITIONAL HOSPITALIZATION REQUIRED? _____
22. SURGERY (IF ANY, DESCRIBE TYPE) _____
23. DATE SURGERY PERFORMED (MO., DAY, YEAR) _____
24. *Press <Return> for OTHER TREATMENT PROVIDED*
25. *Press <Return> for PERMANENT EFFECTS*
26. DATE OF FIRST EXAMINATION (MO., DAY, YEAR) _____
27. Press <Return> for TREATMENT DATES _____
28. DATE DISCHARGED FROM TREATMENT (MO., DAY, YEAR) _____
29. PERIOD OF DISABILITY
TOTAL DISABILITY: FROM _____ TO _____
PARTIAL DISABILITY: FROM _____ TO _____
30. DATE EMPLOYEE ABLE TO RESUME, LIGHT WORK _____
RESUME REGULAR WORK _____

31. EMPLOYEE ADVISED RESUME TO WORK? _____ IF YES, FURNISH DATE _____
32. *Press <Return> for WORK LIMITATIONS*
33. *Press <Return> for REMARKS*
34. DOES DOCTOR SPECIALIZE? _____ SPECIALITY _____
35. ATTENDING PHYSICIAN _____
36. ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
37. TAX IDENTIFICATION NUMBER _____ 38. DATE OF REPORT _____

TREATMENT DATES _____

DATE OF TREATMENT _____
DATE OF TREATMENT _____
DATE OF TREATMENT _____

I. PRE-FORM CODE

n file, rptfil, filing s: 'Sd(nocomp) file=ptc(iform,99)
 +s: 'Sd(nocomp) file=ptg(903,iform,99) s New=0, rptfile=1, idedist=2 k ident
 s U="" i 'Sd(DT) s X= 'Sd(SH), %DT="" d %DT s DT=Y k X,Y
 d sec finfm

II. WINDOWS

WI AUTHORIZATION

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
1604 MEDICAL CARE FACILITY	FREE TEXT	K:SL(X)>30!(SL(X)<1) X
M/FORM FID # 49		
28 STORED: Sp('ICC(DA,1601),"",1)		
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s: 'Sd(X) cm="" s:cm("" cm=""		
1604.1 MED CARE STREET ADDRESS	FREE TEXT	K:SL(X)>54!(SL(X)<1) X
M/FORM FID # 52		
28 STORED: Sp('ICC(DA,1601),"",2)		
44 POST ID: s X=cm K:SL(X)>54!(SL(X)<1) X s: 'Sd(X) cm="" s:cm("" cm=""		
1604.2 MED CARE CITY	FREE TEXT	K:SL(X)>20!(SL(X)<1) X
M/FORM FID # 50		
28 STORED: Sp('ICC(DA,1601),"",3)		
44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s: 'Sd(X) cm="" s:cm("" cm=""		
1604.3 MED CARE STATE	POINTER 1003' DIZ(1003,	
M/FORM FID # 51		
20 CONVERSION CODE: s cm= \$s(cm="" "" , 'Sd('DIZ(1003,cm,0))#10):cm,1:Sp('DIZ(1003,cm,0),"",1))		
28 STORED: Sp('ICC(DA,1601),"",4)		
1604.4 MED CARE ZIP CODE	FREE TEXT	K:SL(X)>10!(SL(X)<5) X
M/FORM FID # 53		
28 STORED: Sp('ICC(DA,1601),"",5)		
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s: 'Sd(X) cm="" s:cm("" cm=""		
2 NAME	POINTER 1004' EMPLOY	
M/FORM FID # 79		

17 EXTERNAL DEFAULT: NAME
 18 INTERNAL DEFAULT: NAMEPTR
 20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d('EMPLOY(cm,0))#10):cm,1:\$p('EMPLOY(cm,0),"",1))
 28 STORED: \$p('ICC(DA,0),"",3)
 32 POST FILING CODE: d setindx^fmfm
 44 POST ID: i \$d(new) s:'\$d('ICC("C",new,x1)) msg="Claim no logged",y=24,cm=""
 54 INDEX: 'ICC("C")
 55 LOOKUP NODE:
 79^45\$~\$p('ICC(lids,0),"",3)
 3^23\$~\$p('ICC(lids,0),"",2)

1 INJURY/ILLNESS DATE & TIME DATE S NOT="ET" D "NOT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 3

20 CONVERSION CODE:
 s X="(" s:+\$s(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +\$s(cm,4,5)) " " s:+\$s(cm,6,7) X=X \$s(cm,6,7) ", " s X=X (1700+\$s(cm,1,3))
 s:\$p(cm,".",2)'=" X=X " at " \$s(cm"000",9,10) ":" \$s(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X
 28 STORED: \$p('ICC(DA,0),"",2)
 32 POST FILING CODE: d setindx^fmfm
 54 INDEX: 'ICC("D")
 55 LOOKUP NODE:
 3^23\$~\$p('ICC(lids,0),"",2)
 79^45\$~\$p('ICC(lids,0),"",3)

15 OCCUPATION FOUNDER 1001' DIZ(1001,

M/FORM FID # 4

20 CONVERSION CODE: s cm=\$s(cm="":",',(\$d('DIZ(1001,cm,0))#10):cm,1:\$p('DIZ(1001,cm,0),"",2))
 28 STORED: \$p('ICC(DA,3),"",1)
 32 POST FILING CODE: d setindx^fmfm

101.1 NATURE OF INJURY EXPLAN

M/FORM FID # 5

28 STORED: nowhere
 29 PREFIELD CODE:
 s:\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d filecns^ptrfilel k shhfile
 i \$d(file),file,new=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
 +d upcntr^fmfm s New=0,cbi(0)=" k file
 s lvl=lvl+1
 i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=CA(i-1)
 30 POSTFIELD CODE:
 i lvl s DA(0)=CA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=CA(i)
 s:lvl DA=CA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1

16 CALLS M/FORM: NATURE OF INJURY/ILLNESS (#642)

M/FORM: NATURE OF INJURY/ILLNESS (#642)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm
```

II. WINDOWS

NATURE OF INJURY/ILLNESS Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 NATURE OF INJURY EXPLAN	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(`ICC(DA(1),101,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm
```

1601 TREATMENT/EXAM FOR CAUSE	SET OF CODES	A:AGREE TO FEE SCHEDULE B:FURNISH NECESSARY TREATMENT C:EXAM FOR JOB RELATED CAUSE &
-------------------------------	--------------	--

M/FORM FID # 6

20 CONVERSION CODE: s cm=\$p(\$p("A:AGREE TO FEE SCHEDULE;B:FURNISH NECESSARY TREATMENT NOT SURGERY;
C:EXAM FOR JOB RELATED CAUSE & TREATMENT;",";"_cm ":"2),"1),cm=\$p(cm,"1)
28 STORED: \$p(`ICC(DA,1600),"",1)

1623 AUTHORIZING OFFICIAL NAM	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
-------------------------------	-----------	--------------------------

M/FORM FID # 77

28 STORED: \$p(`ICC(DA,1610),"",1)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

1623 AUTHORIZING OFFICIAL TIT	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
-------------------------------	-----------	--------------------------

M/FORM FID # 78

28 STORED: \$p(`ICC(DA,1610),"",2)
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

402 SUPERVISOR'S NAME	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
-----------------------	-----------	--------------------------

M/FORM FID # 45

28 STORED: \$p(^ICC(DA,402), "", 1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" cm=""

402.1 SUPERVISOR'S TITLE FREE TEXT K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 46

28 STORED: \$p(^ICC(DA,402), "", 2)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

402.2 SUPERVISOR'S PHONE NUMBER FREE TEXT K:\$L(X)>14!(\$L(X)<7) X

M/FORM FID # 48

28 STORED: \$p(^ICC(DA,402), "", 3)

44 POST ID: s X=cm K:\$L(X)>14!(\$L(X)<7) X s:'\$d(X) cm="" s:cm[""] cm=""

402.4 DATE SUPERVISOR SIGNED DATE S \$DT="E" D ^TOT S X=Y K:Y<1 X

M/FORM FID # 47

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC", ".
+Se(cm,4,5)) " " s:+\$e(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " _Se(cm,"000",9,10) _:" _Se(cm,"000",11,12) s X=X _"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,402), "", 5)

14 REPORTING OFFICE FOURIER 1063' INU(1063,

M/FORM FID # 72

20 CONVERSION CODE: s cm=\$e(cm:"", "", '(\$d(^INU(1063,cm,0))#10):cm,1:\$p(^INU(1063,cm,0), "", 1))

28 STORED: \$p(^ICC(DA,10), "", 1)

14.01 PO ADDRESS FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 74

28 STORED: \$p(^ICC(DA,10), "", 2)

44 POST ID: s X=cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

14.02 PO CITY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 75

28 STORED: \$p(^ICC(DA,10), "", 3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""

14.03 PO STREET FOURIER 1003' DIEZ(1003,

M/FORM FID # 73

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))
28 STORED: \$p('IOC(DA,10),"",4)

14.04 PO ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 76

28 STORED: \$p('IOC(DA,10),"",5)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

W2 PHYSICIAN'S REPORT

DATA ELEMENT DATA TYPE POINTER GLOBAL REFERENCE OR INPUT VERIFICATION

13 DUTY STATION POINTER 1062' INU(1062,

M/FORM FID # 39

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d('INU(1062,cm,0))#10):cm,1:\$p('INU(1062,cm,0),"",1))
28 STORED: \$p('IOC(DA,9),"",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 40

28 STORED: \$p('IOC(DA,9),"",5)
44 POST ID: s X=cm K:\$L(X)>70!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.5 DUTY STATION CITY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 41

28 STORED: \$p('IOC(DA,9),"",6)
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

13.6 DUTY STATION STATE POINTER 1003' DIZ(1003,

M/FORM FID # 42

20 CONVERSION CODE: s cm=\$s(cm="": "", '(\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))
28 STORED: \$p('IOC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 43

28 STORED: \$p('IOC(DA,9),"",8)
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

1605 HISTORY EMPLOYEE GAVE DO

M/FORM FID # 11

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone`ptrfile1 k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,l:filing)
+d upcntr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: HISTORY GIVEN BY EMPLOYEE (#701)

M/FORM: HISTORY GIVEN BY EMPLOYEE (#701)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=`ptc(ifcom,99)
+:$d(nocomp) filen=`ptg(903,ifcom,99) s New=G,rptfile=1
s U="" i '$d(DT) s X=$d($H),MDT="" d `MDT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm
```

II. WINDOWS

HISTORY GIVEN BY EMPLOYEE Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 HISTORY EMPLOYEE GAVE DO	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(`IOC(DA(1),1602,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'>filing:lvl,l:filing)
+d shuffle`fmfm d node0`fmfm
```

1606 PRE/PRIOR CONDITION	SET OF CODES	Y:YES N:NO
--------------------------	--------------	---------------

M/FORM FID # 85

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"_"_":",2),"",1),cm=\$p(cm,"",1)

28 STORED: \$p(`IOC(DA,1600),"",3)

1606.1EXPLAIN INJ/DIS/IMP

M/FORM FID # 13

28 STORED: nowhere

29 PREFIELD CODE:

```
I '$d(cbi(85)) s skip="D"
I '$d(cbi(85)) s:cbi(85){"N"}{"n"} skip="D"
s:'$d(lvl) lvl=0 i '$d(cbi(0))>9 s sthfile=1 d fileone~ptrfile1 k sthfile
i '$d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=1:1:lvl s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1
```

16 CALLS M/FORM: DESCRIBE CONDITION (#702)

M/FORM: DESCRIBE CONDITION (#702)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d "%DT s DT=Y k X,Y
d sec~fufm
d crink~fufm
```

II. WINDOWS

DESCRIBE CONDITION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN IN/DIS/IMP	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),1603,DA(0),""),1)

III. POST-FORM CODE

```
i '$d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle~fufm d node0~fufm
```

1707 DESCRIBE CLINICAL FINDIN

M/FORM FID # 14

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i '$d(cbi(0))>9 s sthfile=1 d fil-one~ptrfile1 k sthfile
i '$d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
```

i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),Dl s lvl=lvl-1

16 CALLS M/FORM: DESCRIBE CLINICAL FINDING (#693)

M/FORM: DESCRIBE CLINICAL FINDING (#693)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=^ptc(iform,99)
+s:\$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$zd(\$H),%DT="" d ^DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm

II. WINDOWS

DESCRIBE CLINICAL FINDING Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DESCRIBE CLINICAL FINDIN	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),1702,DA(0)),",",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

1708 DIAGNOSTS (INQUIRY)

M/FORM FID # 15

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(chi(0))>9 s shhfile=1 d fileone^ptrfilel k shhfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,chi(0)= "" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),Dl s lvl=lvl-1

16 CALLS M/FORM: DIAGNOSIS OF CONDITION (#694)

M/FORM: DIAGNOSIS OF CONDITION (#694)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen~ptc(iform,99)
+s:$d(nocomp) filen~ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d ~%DT s DT=Y k X,Y
d sec~fmrn
d crindx~fmrn
```

II. WINDOWS

DIAGNOSIS OF CONDITION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DIAGNOSIS (INQUIRY)	WORD PROCESSING	
-------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p("ICC(DA(1),1703,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle~fmrn d node0~fmrn
```

1607 DOC BELIEVE INU WORK-REL	SET OF CODES	Y:YES N:NO
-------------------------------	--------------	---------------

M/FORM FID # 80

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";","_cm_":",2"),",",1),cm=\$p(cm,"",1)
28 STORED: \$p("ICC(DA,1600),"",5)

1607.1DOC EXPLAIN-NOT WORK REL

M/FORM FID # 17

28 STORED: nowhere

29 POSTFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d filecne~ptrfilel k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcstr~fmrn s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-DA,DO-DA f i=lvl-1:1 s:$d(DA(i-1)) DA(i)-DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)-DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1
```

16 CALLS M/FORM: DOCTOR'S EXPLANATION (#703)

M/FORM: DOCTOR'S EXPLANATION (#703)

I. PRE-FORM CODE

```
n file,rptfile,New s:'$d(nocomp) file=ptc(iform,99)
+s:$d(nocomp) file=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=Earl(SH),WDT="" d WDT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm
```

II. WINDOWS

DOCTOR'S EXPLANATION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DOC EXPLAIN-NOT WORK REL	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: Sp(^ICC(DA(1),1604,DA,0),",",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

WS PART B CONT

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

1608 REQUIRE HOSPITALIZATION	SET OF CODES	Y:YES N:NO
------------------------------	--------------	---------------

M/FORM FID # 81

20 CONVERSION CODE: s cm=Sp(\$p("Y:YES;N:NO",";","_cm_":",2"),",",1),cm=Sp(cm,"",1)

28 STORED: Sp(^ICC(DA,1600),",",4)

1608.1ADMISSION DATE	DATE	S WDT="E" D WDT S X=Y K:Y<1 X
----------------------	------	-------------------------------

M/FORM FID # 19

20 CONVERSION CODE:

```
s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC,"",")
,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1:00+Se(cm,1,3))
s:Sp(cm,"",2)'="" X=X " at " _Se(cm^"000",9,10)'":_Se(cm^"000",11,1 , s X=X ")"
s:X="(1700)" X="" s cm=X
```

28 STORED: Sp(^ICC(DA,1600),",",6)

29 PREFIELD CODE:

I 'Sd(cbi(81)) s skip="D"
 i 'Sd(cbi(81)) s:cbi(81){"N"!(cbi(81){"h"} skip="D"

1608.2 DISCHARGE DATE

DATE

S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 20

20 CONVERSION CODE:

s X="(" s:+Se(am,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +Se(am,4,5)) " " s:+Se(am,6,7) X=X Se(am,6,7) " " s X=X (1700+Se(am,1,3))
 s:Sp(am,".",2)'=" X=X " at " Se(am"000",9,10) ":" Se(am"000",11,12) s X=X ")"
 s:X="(1700)" X=" s amX

28 STORED: Sp("ICC(DA,1600),""",7)

29 PREFIELD CODE:

i 'Sd(cbi(81)) s skip="D"
 i 'Sd(cbi(19)) s skip="D"
 i 'Sd(cbi(81)) s:cbi(81){"N" skip="D" i 'Sd(skip) s:cbi(19)="" skip="D"

1609 ADDITIONAL HOSPITALIZATI

SET OF CODES

Y:YES

N:NO

M/FORM FID # 82

20 CONVERSION CODE: s amSp(Sp("Y:YES;N:NO;",";","_am_":",2)",";",".1),amSp(am,".",1)
 28 STORED: Sp("ICC(DA,1600),""",8)

1610 SURGERY REQUIRED

FREE TEXT

K:SL(X)>30!(SL(X)<2) X

M/FORM FID # 22

28 STORED: Sp("ICC(DA,1600),""",9)

44 POST ID: s X=am K:SL(X)>30!(SL(X)<2) X s:'Sd(X) am="" s:am(" am=""

1611 SURGERY DATE

DATE

S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 23

20 CONVERSION CODE:

s X="(" s:+Se(am,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 +Se(am,4,5)) " " s:+Se(am,6,7) X=X Se(am,6,7) " " s X=X (1700+Se(am,1,3))
 s:Sp(am,".",2)'=" X=X " at " Se(am"000",9,10) ":" Se(am"000",11,12) s X=X ")"
 s:X="(1700)" X=" s amX

28 STORED: Sp("ICC(DA,1600),""",10)

29 PREFIELD CODE:

i 'Sd(cbi(22)) s skip="D"
 i 'Sd(cbi(22)) s:cbi(22)="" skip="D"

1612 OTHER TREATMENT PROC USED

M/FORM FID # 24

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s sthfile=l d fileone'ptrfilel k sthfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr'fntm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: OTHER TREATMENT PROVIDED (#704)

M/FORM: OTHER TREATMENT PROVIDED (#704)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocmp) filen='ptc(iform,99)
+e:$d(nocmp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),WDT="" d 'WDT s DT=Y k X,Y
d sec'fntm
d crinck'fntm
```

II. WINDOWS

OTHER TREATMENT PROVIDED Window I

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

01 OTHER TREATMENT PROVIDED	WORD PROCESSING	
-----------------------------	-----------------	--

M/FORM FID # 1

28 STORED: Sp('IOC(DA(1),1605,DA(0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle'fntm d node0'fntm
```

1613 PERMANENT EFFECTS

M/FORM FID # 25

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s sthfile=l d fileone'ptrfilel k sthfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr'fntm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: PERMANENT EFFECTS (#705)

M/FORM: PERMANENT EFFECTS (#705)

I. PRE-FORM CODE

n file,rptfile,New s:\$d(nocomp) filer=ptc(iform,99)
+s:\$d(nocomp) filer=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(\$i),%DT="" d %DT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm

II. WINDOWS

PERMANENT EFFECTS Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 PERMANENT EFFECTS	WORD PROCESSING	
-----------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),1606,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

1614 DATE OF 1ST EXAM	DATE	S NOT="E" D %DT S X=Y K:Yd X
-----------------------	------	------------------------------

M/FORM FID # 26

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at "\$e(cm,"000",9,10) ":" \$e(cm,"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p('ICC(DA,1600)," ",11)

1615 TREATMENT DATES	DATE
----------------------	------

M/FORM FID # 27

28 STORED: nowhere

29 PREFIELD CODE:

s:\$d(lvl) lvl=0 i \$d(dci(0))>9 s shhfile=1 d filecne`ptrfilel k shhfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)


```
+d upontr`fmfm s New=0,obi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,CO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: TREATMENT DATES (#706)

M/FORM: TREATMENT DATES (#706)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=`ptc(ifcom,99)
+s:$d(nocomp) filen=`ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),DT="" d `DT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm
```

II. WINDOWS

TREATMENT DATES Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 TREATMENT DATES	DATE	S DT="E" D `DT S X=Y R:Y<1 X

M/FORM FID # 1

20 CONVERSION CODE:

```
s X="(" s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " $e(cm,"000",9,10) ":" $e(cm,"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X
```

28 STORED: \$p(`IOC(DA(1),1607,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$e('$d(filing):lvl,lvb>filing:lvl,1:filing)
+d node0`fmfm
```

WA PART B

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
1616 DATE DISCHARGED FROM TRE	DATE	S DT="E" D `DT S X=Y R:Y<1 X

M/FORM FID # 28

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'=" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,1600),"",12)

1713 TOTAL DISABILITY FROM DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 68

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'=" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,1707),"",1)

1713.1 TOTAL DISABILITY TO DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 69

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'=" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,1707),"",2)

44 POST ID:

i 'Sd(cbi(68)) s skip="D"
 s X=cm k:XcSp(cbi(68),cb,2) X i 'Sd(X) s cm="" s:cm[""" cm=""

1714 PARTIAL DISABILITY FROM DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 70

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'=" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,1707),"",3)

1714.1 PARTIAL DISABILITY TO DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 71

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))

s:Sp(cm,".",2)'="" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,1707),"",4)

44 FOST IL:

i 'Sd(chi(70)) s skip="D"

s X=cm k:XcSp(chi(70),db,2) X i 'Sd(X) s cm="" s:cm["" cm=""

1617 DATE RESUME LIGHT WORK DATE S NOT="E" D ^NOT S X=Y R:Y<1 X

M/FORM FID # 31

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,1600),"",13)

1618 DATE RESUME REG WORK DATE S NOT="E" D ^NOT S X=Y R:Y<1 X

M/FORM FID # 32

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^ICC(DA,1600),"",14)

1715 ADVISED EMP CAN RESUME W SET OF CODES Y:YES
N:NO

M/FORM FID # 83

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO;",";" _cm ":"2);","1),cm=Sp(cm,":",1)

28 STORED: Sp(^ICC(DA,1700),"",11)

1715.1DATE OF ADVICE DATE S NOT="E" D ^NOT S X=Y R:Y<1 X

M/FORM FID # 34

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

20 STORED: Sp(^ICC(DA,1700),"",12)

29 PREFIELD CODE:

I 'Sd(chi(83)) s skip="D"

I Sd(chi(83)) s:chi(83){^N skip="D"

1619 EXPLAIN EMP WORK LIMITAT

M/FORM FID # 35

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d filecne^ptrfilel k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: WORK LIMITATIONS (#707)

M/FORM: WORK LIMITATIONS (#707)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=^ptc(iform,99)
+:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s %=$d($H),%DT="" d ^DT s DT=Y k X,Y
d sec^fufm
d crink^fufm
```

II. WORDS

WORK LIMITATIONS window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN EMP WORK LIMITAT	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),1608,DA,0),",",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle^fufm d node0^fufm
```

54 REMARKS

M/FORM FID # 36

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d filecne^ptrfilel k shhfile
i $d(file),file,New'=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
```

```
+d upcntr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1
```

16 CALLS M/FORM: REMARKS (#606)

M/FORM: REMARKS (#606)

I. PRE-FORM CODE

```
n file,rptfile,New s:$d(nocomp) file="ptc(iform,99)
+s:$d(nocomp) file="ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d "%DT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm
```

II. WINDOWS

REMARKS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 REMARKS

WORD PROCESSING

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),8,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$d('$d(filing):lvl,lvb>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm
```

400.6 DOES DOCTOR SPECIALIZE

SET OF CODES

Y:YES
N:NO

M/FORM FID # 65

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",";","_cm_":",2),";",1),cm=\$p(cm,";",1)
28 STORED: \$p(^ICC(DA,400),"",7)

400.7 SPECIALITY

FREE TEXT

K:\$L(X)>30!((\$L(X)<1) X

M/FORM FID # 64

28 STORED: \$p(^ICC(DA,400),"",8)

29 PREFIELD CODE:

```
I '$d(cbi(65)) s skip="D"
s:cbi(65)/("Y" skip="D"
```

i \$d(chi(65)) s:chi(65) ["n"! (chi(65) ["N") skip="D"

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400 ATTENDING PHYSICIAN

FREE TEXT

K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 59

28 STORED: \$p("ICC(DA,400),"",1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.1 DOCS ADDRESS

FREE TEXT

K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 60

28 STORED: \$p("ICC(DA,400),"",2)

44 POST ID: s X=cm K:\$L(X)>45!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.2 DOCS CITY

FREE TEXT

K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 61

28 STORED: \$p("ICC(DA,400),"",3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.3 DOCS STATE

POINTER 1003' DIZ(1003,

M/FORM FID # 62

20 CONVERSION CODE: s cm=Se(cm="" : "", '(\$d("DIZ(1003,cm,0))#10):cm,1:\$p("DIZ(1003,cm,0),"",1))

28 STORED: \$p("ICC(DA,400),"",4)

400.4 DOCS ZIPCODE

FREE TEXT

K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 63

28 STORED: \$p("ICC(DA,400),"",5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm["" cm=""

400.8 TAX ID

FREE TEXT

K:\$L(X)>10!(\$L(X)<1) X

M/FORM FID # 66

28 STORED: \$p("ICC(DA,400),"",9)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<1) X s:'\$d(X) cm="" s:cm["" cm=""

400.9 DATE SIGNED

DATE

S WDT="E" D WDT S X=Y K:Y<1 X

M/FORM FID # 67

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC"," ",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:\$p(cm,".",2)'="" X=X " at " Se(cm "000",9,10) " : " Se(cm "000",11,12) s X=X ")"

s:X="(1700)" X="" s cm=X

28 STORED: Sp(ICC(DA,400),"",10)

III. POST-FORM CODE

i \$d(file),file,New=0 s:\$d(lvl) lvl=0
+s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,l:filing)
+d upcntr^fmfm
i file=1!(\$d(ERFLAG)) d T3CFILE

CA-17 DUTY STATUS REPORT

PART A--SUPERVISOR

1. MEDICAL CARE FACILITY
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
2. OWCP FILE NUMBER _____ 3. NAME _____
4. DATE OF INJURY (MO., DAY, YEAR) _____
5. SOCIAL SECURITY NUMBER _____
6. OCCUPATION _____
7. BODY PART AFFECTED _____
 DESCRIBE HOW INJURY OCCURRED _____
 Press <Return> for NATURE OF INJURY/ILLNESS
8. Press <Return> for USUAL WORK ACTIVITY
9. JOB REQUIRE DRIVING A VEHICLE? _____ SPECIFY VEHICLE _____
 OPERATING MACHINERY? _____ SPECIFY MACHINERY _____
10. THE EMPLOYEE WORKS HOURS PER DAY _____ NUMBER OF DAYS PER WEEK _____
11. REPORTING OFFICE _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
12. DUTY STATION _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____

PART B--PHYSICIAN

- 13A. HISTORY OF INJURY CORRESPOND TO ITEM 7?
 IF NOT, *Press <Return> for EXPLAIN DISCREPENCY*
- 13B. *Press <Return> for DESCRIBE CLINICAL FINDING*
- 13C. DUE TO INJURY _____ *Press <Return> for DIAGNOSIS OF CONDITION*
- 13D. OTHER DISABLING CONDITIONS *Press <Return> for DIAGNOSIS OF OTHER
14. EMPLOYEE ABLE TO PERFORM REGULAR WORK?
 IF SO, FULL-TIME OR PART-TIME _____ PART-TIME HOURS PER DAY _____
15. IF #14 WAS NO, Press <Return> for ALTERED WORK ACTIVITY
16. *Press <Return> for MEDICAL WORK RESTRICTIONS*
17. PERIOD OF TOTAL DISABILITY: FROM _____ TO _____
 PARTIAL DISABILITY: FROM _____ TO _____
18. EMPLOYEE ADVISED CAN RESUME WORK? _____ DATE OF ADVICE _____
19. DATE OF EXAMINATION _____
20. DATE OF NEXT APPOINTMENT, IF SCHEDULED _____
21. ATTENDING PHYSICIAN _____
 DOCS ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
22. SPECIALITY _____
23. TAX ID NUMBER _____ DATE SIGNED _____

USUAL WORK ACTIVITY

ACTIVITY/EXPOSURE

CONTINUOUS/INTERMITTENT _____ HOURS PER DAY _____ DBA _____ DEGREES F _____
 DESCRIPTION OF ACTIVITY _____

CA-17 CON'T

ALTERED WORK ACTIVITY

ACTIVITY/EXPOSURE

CONTINUOUS/INTERMITTENT _____ HOURS PER DAY _____ DBA _____ DEGREES F _____

DESCRIBE ACTIVITY _____

*Word Processing Window will appear.

M/FORM: CA-17 (#689)

I. PRE-FORM CODE

n filen,rptfile,filing s:'\$d(nocomp) filen='ptc(iform,99)
 +s:\$d(nocomp) filen='ptg(903,iform,99) s N=0,rptfile=1,ident=2 k ident
 s U="" i '\$d(DT) s X=\$d(SH),DT="" d 'DT s DT=Y k X,Y
 d sec'fmfm

II. WINDOWS

WI SUPERVISOR

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
1604 MEDICAL CARE FACILITY	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
M/FORM FID # 39		
28 STORED: \$p('ICC(DA,1601),"",1)		
44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""		
1604.1MED CARE STREET ADDRESS	FREE TEXT	K:\$L(X)>54!(\$L(X)<1) X
M/FORM FID # 36		
28 STORED: \$p('ICC(DA,1601),"",2)		
44 POST ID: s X=cm K:\$L(X)>54!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""		
1604.2MED CARE CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
M/FORM FID # 35		
28 STORED: \$p('ICC(DA,1601),"",3)		
44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""		
1604.3MED CARE STATE	POINTER 1003' DIZ(1003,	
M/FORM FID # 37		
20 CONVERSION CODE: s cm=\$s(cm="" "" ,('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),"",1))		
28 STORED: \$p('ICC(DA,1601),"",4)		
1604.4MED CARE ZIP CODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
M/FORM FID # 38		
28 STORED: \$p('ICC(DA,1601),"",5)		
44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""		
137 GACP FILE NUMBER	FREE TEXT	K:\$L(X)>13!(\$L(X)<1) X
M/FORM FID # 2		

28 STORED: \$p(^ICC(DA,130),"",6)
 44 POST ID: s X=cm K:SL(X)>13!(SL(X)<1) X s:'\$d(X) cm="" s:cm "" cm=""

2 NAME

POINTER 1004' EMPLOY(

M/FORM FID # 62

17 EXTERNAL DEFAULT: NAME
 18 INTERNAL DEFAULT: NAMEPIR
 20 CONVERSION CODE: s cm=\$e(cm:"":'(\$d(^EMPLOY(cm,0))#10):cm,1:\$p(^EMPLOY(cm,0),"",1))
 28 STORED: \$p(^ICC(DA,0),"",3)
 32 POST FILING CODE: d setindx^fmfm
 44 POST ID: i \$d(new) s:'\$d(^ICC("C",new,x1)) msg="Claim not logged",y=24,cm=""
 54 INDEX: ^ICC("C")
 55 LOOKUP NODE:
 62^30^\$p(^ICC(Lids,0),"",3)
 4^23^\$p(^ICC(Lids,0),"",2)

1 INJURY/ILLNESS DATE & TI DATE

S %DT="ET" D ^DT S X=Y K:Y<1 X I \$D(X) K:\$P(X,".",1)>DT X

M/FORM FID # 4

20 CONVERSION CODE:
 s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 +\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
 s:\$p(cm,".",2)'="" X=X " at "\$e(cm,"000",9,10)'": "\$e(cm,"000",11,12) s X=X "
 s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA,0),"",2)
 32 POST FILING CODE: d setindx^fmfm
 54 INDEX: ^ICC("D")
 55 LOOKUP NODE:
 4^23^\$p(^ICC(Lids,0),"",2)
 62^30^\$p(^ICC(Lids,0),"",3)

3 SSN

FREE TEXT

K:SL(X)>9!(SL(X)<9)!'(X?9N) X

M/FORM FID # 5

28 STORED: \$p(^ICC(DA,0),"",4)
 44 POST ID: s X=cm K:SL(X)>9!(SL(X)<9)!'(X?9N) X s:'\$d(X) cm="" s:cm "" cm=""

15 OCCUPATION

POINTER 1001' DEZ(1001,

M/FORM FID # 6

20 CONVERSION CODE: s cm=\$e(cm:"":'(\$d(^DEZ(1001,cm,0))#10):cm,1:\$p(^DEZ(1001,cm,0),"",2))
 28 STORED: \$p(^ICC(DA,3),"",1)
 32 POST FILING CODE: d setindx^fmfm

100 BODY PART CODE

POINTER 1018' INJ(1018,

M/FORM FID # 7

20 CONVERSION CODE: s cm=Sa(cm="":",('Sd('INV(1018,cm,0))%10):cm,1:Sp('INV(1018,cm,0),"",2))
 28 STORED: Sp('IOC(DA,100),"",1)

101.1 NATURE OF INJURY EXPLANA

M FORM FID # 8

28 STORED: nowhere

29 PREFIELD CODE:

s:'Sd(lvl) lvl=0 i Sd(cbi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
 i Sd(file),file,New=0 s filing=Sa('Sd(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
 +d upcntr^fnfma s New=0,cbi(0)=" k file
 s lvl=lvl+1
 i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:Sd(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:Sd(DA(i)) DA(i-1)=DA(i)
 s:lvl DA=DA(0) k DA(lvl),DA(0),Dl s lvl=lvl-1

16 CALLS M FORM: NATURE OF INJURY/ILLNESS (#642)

M FORM: NATURE OF INJURY/ILLNESS (#642)

I. PRE-FORM CODE

n filen,rptfile,New s:'Sd(nocomp) filen='ptc(iform,99)
 +s:Sd(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
 s U=" i 'Sd(DT) s X=Sad(SH),NOT=" d "NOT s DT=Y k X,Y
 d sec^fnfma
 d crinck^fnfma

II. WINDOWS

NATURE OF INJURY/ILLNESS Window 1

DATA ELEMENT	DATA TYPE	FOEMIER GLOBAL REFERENCE OR INPUT VERIFICATION
101 NATURE OF INJURY EXPLANA	WORD PROCESSING	

M FORM FID # 1

28 STORED: Sp('IOC(DA(1),101,DA,0),"",1)

III. POST-FORM CODE

i Sd(file),file s filing=Sa('Sd(filing):lvl,lvl>filing:lvl,1:filing)
 +d shufFile^fnfma d node0^fnfma

1720 USUAL WORK ACTIVITY	FOEMIER
--------------------------	---------

M FORM FID # 73

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shfile=l d fileone=ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr=fnfcm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DC-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: USUAL WORK ACTIVITY (#691)

M/FORM: USUAL WORK ACTIVITY (#691)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),DT="" d 'NOT s DT=Y k X,Y
d sec=fnfcm
```

II. WINDOWS

USUAL WORK ACTIVITY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 USUAL WORK ACTIVITY	POINTER 1007	INU(1007,

M/FORM FID # 1

```
20 CONVERSION CODE: s cm=$s(cm="":",",'($d('INU(1007,cm,0))#10):cm,1:$p('INU(1007,cm,0),"",2))
28 STORED: $p('ICC(DA(1),1709,DA,0),"",1)
32 POST FILING CODE: d sec=fnfcm
54 INDEX: 'ICC(DA(1),1709,"B")
55 LOOKUP CODE: 1'60'$p('ICC(DA(1),1709,1ids,0),"",1)
```

2 CONTINUER	SET OF CODES	C:CONTINUOUS I:INTERMITTENT
-------------	--------------	--------------------------------

M/FORM FID # 3

```
20 CONVERSION CODE: s cm=$p($p(";C:CONTINUOUS;I:INTERMITTENT;";"_cm_":",2);",1),cm=$p(cm,"":",1)
28 STORED: $p('ICC(DA(1),1709,DA,0),"",3)
```

3 HOURS PER DAY	NUMERIC FORMAT	R:X'-X!(X>10)!(X<0)!(X2.E1".IN.N) X
-----------------	----------------	-------------------------------------

M/FORM FID # 4

```
28 STORED: $p('ICC(DA(1),1709,DA,0),"",4)
```

5 DPA	NUMERIC FORMAT	R:X'-X!(X>150)!(X<0)!(X2.E1".IN.N) X
-------	----------------	--------------------------------------

M/FORM FID # 6

28 STORED: \$p("IOC(DA(1),1709,DA,0),"",6)

29 PREFIELD CODE:

i 'Sd(chi(1)) s skip="D"

i Sd(chi(1)) s:\$p(chi(1),db,1){"NOISE " skip="D"

4 DEGREES F

NUMERIC FORMAT

K:X'-X(1)(X>200)!(X<0)!(X>.E1"."IN.N) X

M/FORM FID # 5

28 STORED: \$p("IOC(DA(1),1709,DA,0),"",5)

29 PREFIELD CODE:

i 'Sd(chi(1)) s skip="D"

i Sd(chi(1)) s:\$p(chi(1),db,1){"TEMP"\$(Sd(chi(1),db,1){"COLD"}) skip="D"

1 DESCRIPTION OF ACTIVITY

FREE TEXT

K:SL(X)>50!(SL(X)<1) X

M/FORM FID # 7

28 STORED: \$p("IOC(DA(1),1709,DA,0),"",2)

44 POST ID: s X-on K:SL(X)>50!(SL(X)<1) X s:'Sd(X) on"" s:cm[""" on""

III. POST-FORM CODE

i Sd(file),file s filing-\$s('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d node0" finfm

1702 JOB REQUIRES DRIVING

SET OF CODES

Y:YES

N:NO

M/FORM FID # 68

20 CONVERSION CODE: s on-\$p(\$p("Y:YES;N:NO","";"on ":"",2),"",1),on-\$p(on,"",1)

28 STORED: \$p("IOC(DA,1700),"",1)

1702.1SPECIFY VEHICLE

FREE TEXT

K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 11

28 STORED: \$p("IOC(DA,1700),"",9)

29 PREFIELD CODE:

i 'Sd(chi(68)) s skip="D"

i Sd(chi(68)) s:chi(68){"n"!(chi(68){"N"}) skip="D"

44 POST ID: s X-on K:SL(X)>20!(SL(X)<1) X s:'Sd(X) on"" s:cm[""" on""

1703 JOB REQ OPERATING MACHIN

SET OF CODES

Y:YES

N:NO

M/FORM FID # 69

20 CONVERSION CODE: s cm=Sp(Sp;"Y: YES;N: NO;",";"_cm_":",2);",1),cm=Sp(cm,"",1)
28 STORED: Sp("ICC(DA,1700),"",2)

1703.1 SPECIFY MACHINERY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 13

28 STORED: Sp("ICC(DA,1700),"",10)

29 PREFIELD CODE:

I 'Sd(cbi(69)) s skip="D"

i Sd(cbi(69)) s:cbi(69)!"n"!(cbi(69)!"n") skip="D"

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

1704 HRS PER DAY EMP WORKS NUMERIC FORMAT K:+X'-X!(X>8)!(X<1)!(X>.E1"."N.N) X

M/FORM FID # 14

28 STORED: Sp("ICC(DA,1700),"",3)

1705 DAYS PER WEEK EMP WORKS NUMERIC FORMAT K:+X'-X!(X>5)!(X<1)!(X>.E1"."N.N) X

M/FORM FID # 15

28 STORED: Sp("ICC(DA,1700),"",4)

14 REPORTING OFFICE POINTER 1063' INJ(1063,

M/FORM FID # 40

20 CONVERSION CODE: s cm=Sp(cm="":",','(Sd("INJ(1063,cm,0))#10):cm,1:Sp("INJ(1063,cm,0),"",1))
28 STORED: Sp("ICC(DA,10),"",1)

14.01 PO ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 41

28 STORED: Sp("ICC(DA,10),"",2)

44 POST ID: s X=cm K:SL(X)>70!(SL(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

14.02 PO CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 42

28 STORED: Sp("ICC(DA,10),"",3)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

14.03 PO STATE POINTER 1003' DIZ(1003,

M/FORM FID # 43

20 CONVERSION CODE: s cm=Sp(cm="":",','(Sd("DIZ(1003,cm,0))#10):cm,1:Sp("DIZ(1003,cm,0),"",1))
28 STORED: Sp("ICC(DA,10),"",4)

14.04 RD ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 44

28 STORED: \$p(^ICC(DA,10),"",5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:\$d(X) cm="" s:cm(" " cm="")

13 DUTY STATION POINIER 1062' INU(1062,

M/FORM FID # 45

20 CONVERSION CODE: s cm=\$s(cm="":",',"(\$d(^INU(1062,cm,0))#10):cm,1:\$p(^INU(1062,cm,0),"",1))

28 STORED: \$p(^ICC(DA,9),"",1)

13.4 DUTY STATION ADDRESS FREE TEXT K:\$L(X)>54!(\$L(X)<1) X

M/FORM FID # 46

28 STORED: \$p(^ICC(DA,9),"",5)

44 POST ID: s X=cm K:\$L(X)>70!(\$L(X)<1) X s:\$d(X) cm="" s:cm(" " cm="")

13.5 DUTY STATION CITY FREE TEXT K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 47

28 STORED: \$p(^ICC(DA,9),"",6)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:\$d(X) cm="" s:cm(" " cm="")

13.6 DUTY STATION STATE POINIER 1003' DIZ(1003,

M/FORM FID # 48

20 CONVERSION CODE: s cm=\$s(cm="":",',"(\$d(^DIZ(1003,cm,0))#10):cm,1:\$p(^DIZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,9),"",7)

13.7 DUTY STATION ZIP CODE FREE TEXT K:\$L(X)>10!(\$L(X)<5) X

M/FORM FID # 49

28 STORED: \$p(^ICC(DA,9),"",8)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:\$d(X) cm="" s:cm(" " cm="")

WZ PHYSICIAN

DATA ELEMENT DATA TYPE POINIER GLOBAL REFERENCE OR INPUT VERIFICATION

1706 DOES DOC AGREE W/ EMP ST SET OF CODES Y:YES
N:NO

M/FORM FID # 71

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;",":",":_cm_":",2),":",1),cm=\$p(cm,"",1)

28 STORED: \$p(^ICC(DA,1700),"",5)

1706.1EXPLAIN WHAT EMP TOLD YO

M/FORM FID # 19

28 STORED: nowhere

29 PREFIELD CODE:

```
I 'SD(cbi(71)) s skip="D"
i $d(cbi(71)) s:cbi(71){"Y"}!(cbi(71){"Y"} skip="D"
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone`ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fnfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1
```

16 CALLS M/FORM: EXPLAIN DISCREPENCY (#692)

M/FORM: EXPLAIN DISCREPENCY (#692)

I. PRE-FORM CODE

```
n file,rptfile,New s:$d(nocomp) file=`ptc(iform,99)
+s:$d(nocomp) file=`ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),WOT="" d "WOT s DT=Y k X,Y
d sec`fnfm
d crinck`fnfm
```

II. WINDOWS

EXPLAIN DISCREPENCY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN WHAT EMP TOLD YO	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('IOC(DA(1),1701,DA(0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fnfm d node0`fnfm
```

1707 DESCRIBE CLINICAL FINDIN

M/FORM FID # 20

28 STORED: nowhere

29 PREFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

```

16 CALLS M/FORM: DESCRIBE CLINICAL FINDING (#693)

M/FORM: DESCRIBE CLINICAL FINDING (#693)

I. PRE-FORM CODE

```

n filen,rptfile,New s:'$d(ncomp) filen=^ptc(iform,99)
+s:$d(ncomp) filen=^ptc(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$z(d$H),DT="" d ^DT s DT=Y k X,Y
d sec^fmfm
d crindk^fmfm

```

II. WINDOWS

DESCRIBE CLINICAL FINDING Window I

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DESCRIBE CLINICAL FINDING	WORD PROCESSING	
-------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p^ICC(DA(1),1702,DA,0),"",1)

III. POST-FORM CODE

```

i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

```

1706 DIAGNOSIS (INJURY)

M/FORM FID # 21

28 STORED: rowhen

29 PREFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)

```

s:lvl DA-DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: DIAGNOSIS OF CONDITION (#694)

M/FORM: DIAGNOSIS OF CONDITION (#694)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$d(nocomp) filen=^ptc(ifcom,99)
+s:\$d(nocomp) filen=^ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '\$d(DT) s %=\$d(\$H),%DT="" d ^DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm

II. WINDOWS

DIAGNOSIS OF CONDITION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DIAGNOSIS (INQUIRY)	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),1703,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

1709 DIAGNOSIS (OTHER COND)

M/FORM FID # 22

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-DA,DO-DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)-DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)-DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: DIAGNOSIS OF OTHER (#695)

M/FORM: DIAGNOSIS OF OTHER (#695)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),%DT="" d '%DT s DT=Y k X,Y
d sec'fmfm
d crinck'fmfm
```

II. WINDOWS

DIAGNOSIS OF OTHER WINDOW 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DIAGNOSIS (OTHER COND)	WORD PROCESSING	
----------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),1704,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle'fmfm d node0'fmfm
```

1710 CAN EMP RETURN TO REG WO	SET OF CODES	Y: YES N: NO
-------------------------------	--------------	-----------------

M/FORM FID # 72

20 CONVERSION CODE: s cm=\$p(\$p("Y: YES;N: NO;"_"_cm":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p('ICC(DA,1700),"",6)

1710.1 FULL OR PART-TIME	SET OF CODES	F: FULL TIME P: PART TIME f: FULL TIME p: PART TIME
--------------------------	--------------	--

M/FORM FID # 24

20 CONVERSION CODE: s cm=\$p(\$p("F: FULL TIME;P: PART TIME;f: FULL TIME;
p: PART TIME;"_"_cm":",2),"",1),cm=\$p(cm,"",1)
28 STORED: \$p('ICC(DA,1700),"",7)
29 PREFIELD CODE:
I '\$d(cbl(72)) s skip="D"
I '\$d(cbl(72)) s: cbl(72){ "N" ! (cbl(72){ "N" } skip="D"

1710.11 PART-TIME HRS PER DAY	NUMERIC FORMAT	K: +X'-X1(XD8)!(X<1)!(X<2)!"IN.N) X
-------------------------------	----------------	-------------------------------------

M/FORM FID # 25

28 STORED: \$p('ICC(DA,1700),"",16)

29 PREFIELD CODE:

```
I '$D(cbi(72)) s skip="D"
i $d(rhi(72)) s:cbi(72){"n"!(cbi(72){"N"} skip="D"
I '$D(cbi(24)) s skip="D"
i $d(cbi(24)) s:cbi(24){"F"!(cbi(24){"F"} skip="D"
```

1721 ALTERED WORK ACTIVITY POINIER

M/FORM FID # 74

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone`ptrfilel k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DL=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1
```

16 CALLS M/FORM: ALTERED WORK ACTIVITY (#696)

M/FORM: ALTERED WORK ACTIVITY (#696)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=`ptc(iform,99)
+s:$d(nocomp) filen=`ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),%DT="" d `DT s DT=Y k X,Y
d sec`fmfm
```

II. WINDOWS

ALTERED WORK ACTIVITY Window I

DATA ELEMENT	DATA TYPE	POINIER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 ALTERED WORK ACTIVITY	POINIER 1007'	INU(1007,
---------------------------	---------------	-----------

M/FORM FID # 1

```
20 CONVERSION CODE: s cm=$s(cm="": "",('$d(`INU(1007,cm,0))#10):cm,1:$p(`INU(1007,cm,0),"",2))
28 STORED: $p(`IOC(DA(1),1710,DA,0),"",1)
32 POST FILING CODE: d setindx`fmfm
54 INDEX: `IOC(DA(1),1710,"B")
55 LOOKUP NODE: 1`60`$p(`IOC(DA(1),1710,Lids,0),"",1)
```

2 CONI/INTER	SET OF CODES	C:CONTINUOUS I:INTERMITTENT
--------------	--------------	--------------------------------

M/FORM FID # 3

20 CONVERSION CODE: s cm=Sp(Sp("C:CONTINUOUS;I:INTERMITTENT;";"_cm_":",2);",1),cm=Sp(cm,"")
 28 STORED: Sp("IOC(DA(1),1710,DA,0),"",3)
 29 PREFIELD CODE: i 'Sd(chi(1)) s skip="D"

3 HOURS PER DAY NUMERIC FORMAT K:+X'=X!(X>10)!(X<0)!(X?.E1". "1N.N) X

M/FORM FID # 4

28 STORED: Sp("IOC(DA(1),1710,DA,0),"",4)

5 DBA NUMERIC FORMAT K:+X'=X!(X>150)!(X<0)!(X?.E1". "1N.N) X

M/FORM FID # 6

28 STORED: Sp("IOC(DA(1),1710,DA,0),"",6)

29 PREFIELD CODE:

i 'Sd(chi(1)) s skip="D"

i Sd(chi(1)) s:Sp(chi(1),db,1)["NOISE" skip="D"

4 DEGREES F NUMERIC FORMAT K:+X'=X!(X>200)!(X<0)!(X?.E1". "1N.N) X

M/FORM FID # 5

28 STORED: Sp("IOC(DA(1),1710,DA,0),"",5)

29 PREFIELD CODE:

i 'Sd(chi(1)) s skip="D"

i Sd(chi(1)) s:Sp(chi(1),db,1)["COLD"&(Sp(chi(1),db,1)["HEAT"]) skip="D"

1 DESCRIBE ACTIVITY FREE TEXT K:\$L(X)>50!(\$L(X)<1) X

M/FORM FID # 7

28 STORED: Sp("IOC(DA(1),1710,DA,0),"",2)

44 POST ID: s %cm K:\$L(X)>50!(\$L(X)<1) X s:'Sd(X) cm="" s:cm["" cm=""

III. POST-FORM CODE

i Sd(file),file s filing=Sp('Sd(filing):lvl,lvl>filing:lvl,1:filing)
 +d node0"fmfm

WB PHYSICIAN

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

1712 DESCRIBE FURTHER RESURIC

M/FORM FID # 27

28 STORED: nowhere

29 PREFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone ptrfile k shhfile
i $d(file),file,New=0 s filing=$e('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

```

16 CALLS M/FRM: MEDICAL WORK RESTRICTIONS (#697)

M/FRM: MEDICAL WORK RESTRICTIONS (#697)

I. PRE-FORM CODE

```

n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),%DT="" d %DT s DT=Y k X,Y
d sec^fmfm
d crindx^fmfm

```

II. WINDOWS

MEDICAL WORK RESTRICTIONS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 DESCRIBE FURTHER RESTRIC	WORD PROCESSING	
------------------------------	-----------------	--

M/FRM FID # 1

28 STORED: \$p(^ICC(DA(1),1706,DA,0),"",1)

III. POST-FORM CODE

```

i $d(file),file s filing=$e('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

```

1713 TOTAL DISABILITY FROM	DATE	S %DT="E" D %DT S X=Y K:Y<1 X
----------------------------	------	-------------------------------

M/FRM FID # 50

20 CONVERSION CODE:

```

s X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",+Se(cm,4,5))_ " s:+Se(cm,6,7) X=X Se(cm,6,7)"_ "
s X=X (1700+Se(cm,1,3))
s:$p(cm,".",2)="" X=X " at "_Se(cm_000",9,10)"_:"_Se(cm_000",11,12) s X=X_"
s:X="(1700)" X="" s cmX

```

28 STORED: \$p(^ICC(DA,1707),"",1)

1713.1TOTAL DISABILITY TO	DATE	S %DT="E" D %DT S X=Y K:Y<1 X
---------------------------	------	-------------------------------

M/FORM FID # 51

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10)'=" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,1707),"",2)

44 POST ID:

i \$d(cbi(50)) s skip="D"
s X=cm k:XSp(cbi(50),cb,2) X i '\$d(X) s cm="" s:cm["" cm=""

1714 PARTIAL DISABILITY FROM DATE S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 52

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10)'=" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,1707),"",3)

1714.1 PARTIAL DISABILITY TO DATE S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 53

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10)'=" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^ICC(DA,1707),"",4)

44 POST ID:

i '\$d(cbi(52)) s skip="D"
s X=cm k:XSp(cbi(52),cb,2) X i '\$d(X) s cm="" s:cm["" cm=""

1715 ADVISED EMP CAN RESUME W SET OF CODES Y:YES
N:NO

M/FORM FID # 67

20 CONVERSION CODE: s cm=Sp(Sp(";Y:YES;N:NO;";"cm":",2),";",1),cm=Sp(cm,";",1)

28 STORED: Sp(^ICC(DA,1700),"",11)

1715.1 DATE OF ADVICE DATE S ^DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 31

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",

,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1700),"",12)

29 PREFIELD CODE:

I 'SD(chi(67)) s skip="D"

I \$d(chi(67)) s:chi(67){ "n"! (chi(67)) {"N"} skip="D"

1701 DATE OF EXAM

DATE

S WDT="E" D ^WDT S X=Y K:Y<1 X

M/FORM FID # 32

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB..MAR.APR..MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1700),"",14)

1716 NEXT EXAM DATE

DATE

S WDT="E" D ^WDT S X=Y K:Y<1 X

M/FORM FID # 33

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB..MAR.APR..MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " _Se(cm_"000",9,10) _:" _Se(cm_"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1700),"",13)

400 ATTENDING PHYSICIAN

FREE TEXT

K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 54

28 STORED: Sp(^IOC(DA,400),"",1)

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

400.1 DOCS ADDRESS

FREE TEXT

K:SL(X)>54!(SL(X)<1) X

M/FORM FID # 55

28 STORED: Sp(^IOC(DA,400),"",2)

44 POST ID: s X=cm K:SL(X)>45!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

400.2 DOCS CITY

FREE TEXT

K:SL(X)>20!(SL(X)<1) X

M/FORM FID # 56

28 STORED: Sp(^IOC(DA,400),"",3)

44 POST ID: s X=cm K:SL(X)>20!(SL(X)<1) X s:'\$d(X) cm="" s:cm(""" cm=""

400.3 DOCS STATE

FOINDER 1003' DIZ(1003,

M FORM FID # 57

20 CONVERSION CODE: s cm=Se(cm="":", '(Sd('DIZ(1003,cm,0))#10):cm,1:Sp('DIZ(1003,cm,0),"",1))
28 STORED: Sp('ICC(DA,400),"",4)

400.4 DOCS LIPODE

FREE TEXT

K:SL(X)>10!(SL(X)<5) X

M FORM FID # 58

28 STORED: Sp('ICC(DA,400),"",5)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'Sd(X) cm="" s:cm(""" cm=""

400.7 SPECIALITY

FREE TEXT

K:SL(X)>30!(SL(X)<1) X

M FORM FID # 59

28 STORED: Sp('ICC(DA,400),"",8)
44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm(""" cm=""

400.8 TAX ID

FREE TEXT

K:SL(X)>10!(SL(X)<1) X

M FORM FID # 60

28 STORED: Sp('ICC(DA,400),"",9)
44 POST ID: s X=cm K:SL(X)>10!(SL(X)<1) X s:'Sd(X) cm="" s:cm(""" cm=""

400.9 DATE SIGNED

DATE

S NOT="E" D NOT S X=Y K:Y<1 X

M FORM FID # 61

20 CONVERSION CODE:
s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",
+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X "
s:X="(1700)" X="" s cm=X
28 STORED: Sp('ICC(DA,400),"",10)

III. POST-FORM CODE

i Sd(file),file,New=0 s:'Sd(lvl) lvl=0
+s filing=Se('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr'filing
i file=1!(Sd(ENFLAG)) d TOFILE

CA-20 ATTENDING PHYSICIAN'S REPORT

RECORD OF EXAMINATION

1. PATIENT'S NAME (LAST, FIRST MIDDLE) _____
2. DATE OF INJURY _____
3. CWCP FILE NUMBER _____
4. PRE-EXISTING INJURY OR DISEASE _____ ICD-9 CODE _____
 Press <Return> for DESCRIBE CONDITION
5. *Press <Return> for DESCRIBE CLINICAL FINDING*
6. *Press <Return> for DIAGNOSIS OF CONDITION* ICD-9 CODE _____
7. EMPLOYMENT CAUSED/AGGRAVATED CONDITION?
 PLEASE EXPLAIN ANSWER *Press <Return> for DOCTOR'S EXPLANATION*
8. DID INJURY REQUIRE HOSPITALIZATION? _____
9. DATE OF ADMISSION _____ 10. DATE OF DISCHARGE _____
11. ADDITIONAL HOSPITALIZATION REQUIRED? _____
12. WHAT TREATMENT DID YOU PROVIDE?
 Press <Return> for OTHER TREATMENT PROVIDED
13. DATE OF FIRST EXAM _____
14. Press <Return> for TREATMENT DATES _____
15. DATE DISCHARGED FROM TREATMENT _____
16. PERIOD OF TOTAL DISABILITY FROM _____ THROUGH _____
17. PERIOD OF PARTIAL DISABILITY FROM _____ THROUGH _____
18. DATE EMPLOYEE ABLE TO RESUME LIGHT WORK _____
19. DATE ABLE TO RESUME REGULAR WORK _____
20. ADVISED EMPLOYEE TO RETURN TO WORK? _____ 21. DATE OF ADVISE _____
22. *Press <Return> for WORK LIMITATIONS* 23. ANY PERMANENT EFFECTS _____
24. *Press <Return> for REMARKS* _____
25. NAME OF PHYSICIAN IF EMP NEED REFER TO _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
 SPECIALITY _____
26. REASON FOR REFERRAL _____
27. SIGNATURE OF PHYSICIAN AND DATE SIGNED _____
28. NAME OF PHYSICIAN _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
29. TAX ID NUMBER _____
30. DO YOU SPECIALIZE? _____
31. IF YES, INDICATE SPECIALITY _____

TREATMENT DATES

DATE OF TREATMENT _____
 DATE OF TREATMENT _____
 DATE OF TREATMENT _____

*Word Processing Window will appear.

I. PRE-FORM CODE

n file, rptfile, filing s: 'Sd(nocomp) file=ptc(ifom,99)
+s:Sd(nocomp) file=ptg(903,ifom,99) s New=0, rptfile=1, idedist=2 k ident
s U="" i 'Sd(DT) s X=Sd(SH), %DT="" d %DT s DT=Y k X,Y
d sec finfm

II. WINDOWS

CA-20 Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

2 NAME	POINTER 1004' EMPLOY	
--------	----------------------	--

M FORM FID # 52

17 EXTERNAL DEFAULT: NAME
18 INTERNAL DEFAULT: NAMEPTR
20 CONVERSION CODE: s cm=Se(cm="" : "", '(Sd(EMPLOY(cm,0))#10):cm,1:Sp(EMPLOY(cm,0),"",1))
28 STORED: Sp(ICC(DA,0),"",3)
32 POST FILING CODE: d setindx finfm
44 POST ID: i Sd(new) s: 'Sd(ICC("C",new,x1)) msg="Claim not logged",y=24,cm=""
54 INDEX: ICC("C")
55 LOOKUP NODE:
52'35'Sp(ICC(Lids,0),"",3)
2'23'Sp(ICC(Lids,0),"",2)

1 DAILY ILLNESS DATE & TI	DATE	S %DT="ET" D %DT S %Y K:Y<1 X I SD(X) K:SP(X,".",1)>DT X
---------------------------	------	--

M FORM FID # 2

20 CONVERSION CODE:
s %="(" s:+Se(cm,4,5) %X Sp("JAN.FEB.MAR.APR..MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+Se(cm,4,5)) " " s:+Se(cm,6,7) %X Se(cm,6,7) ", " s %X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" %X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s %X "
s:%="(1700)" %="" s cmX
28 STORED: Sp(ICC(DA,0),"",2)
32 POST FILING CODE: d setindx finfm
54 INDEX: ICC("D")
55 LOOKUP NODE:
2'23'Sp(ICC(Lids,0),"",2)
52'35'Sp(ICC(Lids,0),"",3)

137 OACP FILE NUMBER	FREE TEXT	K:SL(X)>L31(SL(X)<1) X
----------------------	-----------	------------------------

M FORM FID # 3

28 STORED: Sp(ICC(DA,L30),"",6)
44 POST ID: s %cm K:SL(X)>L31(SL(X)<1) X s: 'Sd(X) cm="" s:cm "" cm=""

1606 PRE-EXISTING CONDITION

SET OF CODES

Y: YES
N: NO

M/FORM FID # 53

20 CONVERSION CODE: s cm=Sp(Sp("Y: YES; N: NO; ", "_cm": ",2), ", ",1), cm=Sp(cm, ", ",1)
28 STORED: Sp("ICC(DA,1600)", "",3)

2040 ICC-9 CODE FOR PRE-EXIST

FREE TEXT

K: SL(X)>6!(SL(X)<6) X

M/FORM FID # 32

28 STORED: Sp("ICC(DA,2006)", "",7)

44 POST ID: s %cm K: SL(X)>6!(SL(X)<6) X s: 'Sd(X) cm="" s: cm("" cm=""

1606.1 EXPLAIN IN/DIS/IMP

M/FORM FID # 5

28 STORED: nowhere

29 PREFIELD CODE:

i 'Sd(cbi(53)) s skip="D"
i 'Sd(cbi(53)) s: cbi(53) ["N"] (cbi(53) ["r"]) skip="D"
s: 'Sd(lvl) lvl=0 i 'Sd(cbi(0))>9 s shhfile=l d filecm=ptrfilel k shhfile
i 'Sd(file), file, New=0 s filing=6s('Sd(filing):lvl+1,lvl'<filing:lvl+1,l:l:filing)
+d upcmr'fcm s New=0, cbi(0)=" k file
s lvl=lvl+1
i lvl s DA(0)=DA, DO=DA f i=lvl:-1:l s: Sd(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:l:lvl s: Sd(DA(i)) DA(i-1)=DA(i)
s: lvl DA=DA(0) k DA(lvl), DA(0), DI s lvl=lvl-1

16 CALLS M/FORM: DESCRIBE CONDITION (#702)

M/FORM: DESCRIBE CONDITION (#702)

I. PRE-FORM CODE

n filen, rptfile, New s: 'Sd(nocomp) filen=ptr(ifcom,99)
+s: Sd(nocomp) filen=ptr(903, ifcom, 99) s New=0, rptfile=1
s D="" i 'Sd(DT) s %Gad(SH), NOT="" d "NOT s DT=Y k X, Y
d sec'fcm
d crick'fcm

II. WINDOWS

DESCRIBE CONDITION Window 1

DATA ELEMENT

DATA TYPE

FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION

01 EXPLAIN IN/DIS/IMP

WORD PROCESSING

M/FORM FID # 1

28 STORED: \$p('IOC(DA(1),1603,DA,0),'',1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fufm d node0`fufm

1707 DESCRIBE CLINICAL FINDIN

M/FORM FID # 7

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$(lvl) lvl=0 i \$(chi(0))>9 s shhfile=l d fileone`ptrfilel k shhfile
i \$d(file),file,New'=0 s filing-\$s('\$(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fufm s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=DA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: DESCRIBE CLINICAL FINDING (#693)

M/FORM: DESCRIBE CLINICAL FINDING (#693)

I. PRE-FORM CODE

n filen,rptfile,New s:'\$(nocomp) filen=`ptc(ifcom,99)
+s:\$d(nocomp) filen=`ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '\$d(DT) s X=\$d(SH),NOT="" d `NOT s DT=Y k X,Y
d asc`fufm
d crindx`fufm

II. WINDOWS

DESCRIBE CLINICAL FINDING Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DESCRIBE CLINICAL FINDIN	WORD PROCESSING	

M/FORM FID # 1

20 STORED: \$p('IOC(DA(1),1702,DA,0),'',1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fufm d node0`fufm

1708 DIAGNOSIS (INQUIRY)

M/FORM FID # 8

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(obi(0))>9 s shhfile=1 d fileone ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,obi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,D0=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),D1 s lvl=lvl-1
```

16 CALLS M/FORM: DIAGNOSIS OF CONDITION (#694)

M/FORM: DIAGNOSIS OF CONDITION (#694)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocamp) filen=`ptc(iform,99)
+s:$d(nocamp) filen=`ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(UT) s X=$d(SH),UT="" d `UT s UT=X k X,Y
d sec`fmfm
d crinch`fmfm
```

II. WINDOWS

DIAGNOSIS OF CONDITION Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DIAGNOSIS (INQUIRY)	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),1703,DA(0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm
```

2040.1ICD-9 CODE FOR DIAGNOSIS FREE TEXT R:SL(X)>6!(SL(X)<6) X

M/FORM FID # 11

28 STORED: \$p('ICC(DA,2006),"",8)

44 POST ID: s X=cm R:SL(X)>6!(SL(X)<6) X s:'\$d(X) cm="" s:cm[""] cm=""

1607 DOC BELIEVE IN WORK-REL GET OF CODES Y:YES

N:NO

M/FORM FID # 54

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO",";","_cm":",2)",";"),1),cm=\$p(cm,";",1)
28 STORED: \$p("IOC(DA,1600)","",5)

1607.1DOC EXPLAIN-NOT WORK REL

M/FORM FID # 10

28 STORED: nowhere

29 PREFILED CODE:

i \$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d fileone ptrfile k shhfile
 i \$d(file),file,new=0 s filling=\$e(\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
 +d upcntr fmfcm s New=0,cbi(0)="" k file
 s lvl=lvl+1
 i lv: s DA(0)=DA,D0=DA f i=lvl:-1:1 s:\$d(DA(i-1)) DP(i)=DA(i-1)

30 POSTFILED CODE:

i lvl s DA(0)=DA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=DA(i)
 s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

16 CALLS M/FORM: DOCTOR'S EXPLANATION (#703)

M/FORM: DOCTOR'S EXPLANATION (#703)

I. PRE-FORM CODE

n filen,rptfile,new s:\$d(nocomp) filen=ptr(ifoun,99)
 +s:\$d(nocomp) filen=ptg(903,ifoun,99) s New=0,rptfile=1
 s U="" i \$d(UT) s X=\$d(SH),NOT="" d "NOT s DT=Y k X,Y
 d sec fmfcm
 d crinck fmfcm

II. WINDOWS

DOCTOR'S EXPLANATION Window I

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DOC EXPLAIN-NOT WORK REL	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p("IOC(DA(1),1604,DA(0))","",1)

III. POST-FORM CODE

i \$d(file),file s filling=\$e(\$d(filing):lvl,lvl'<filing:lvl+1,1:filing)
 +d shuffle fmfcm d node0 fmfcm

1608 REQUIRE HOSPITALIZATION	SET OF CODES	Y:YES
------------------------------	--------------	-------

N:NO

M/FORM FID # 55

20 CONVERSION CODE: s cm=Sp(\$p(";Y:YES;N:NO;";";_cm ":";2);";;1),cm=Sp(cm,"";1)
28 STORED: Sp(^ICC(DA,1600),"",4)

1608.1 ADMISSION DATE DATE S %DT="E" D ^%DT S %Y K:Ycl X

M/FORM FID # 12

20 CONVERSION CODE:

s %="(" s:+Se(cm,4,5) %X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
,+Se(cm,4,5)) " " s:+Se(cm,6,7) %X Se(cm,6,7) " " s %X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" %X " at " _Se(cm^000",9,10) _:" _Se(cm^000",11,12) s %X _)"
s:%="(1700)" %=" s cm=X

28 STORED: Sp(^ICC(DA,1600),"",6)

29 PREFIELD CODE:

I 'Sd(cbi(55)) s skip="D"
i Sd(cbi(55)) s:cbi(55){ "N"! (cbi(55){ "n") skip="D"

1608.2 DISCHARGE DATE DATE S %DT="E" D ^%DT S %Y K:Ycl X

M/FORM FID # 13

20 CONVERSION CODE:

s %="(" s:+Se(cm,4,5) %X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
,+Se(cm,4,5)) " " s:+Se(cm,6,7) %X Se(cm,6,7) " " s %X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" %X " at " _Se(cm^000",9,10) _:" _Se(cm^000",11,12) s %X _)"
s:%="(1700)" %=" s cm=X

28 STORED: Sp(^ICC(DA,1600),"",7)

29 PREFIELD CODE:

I 'SD(cbi(55)) s skip="D"
i Sd(cbi(55)) s:cbi(55){ "N"! (cbi(55){ "n") skip="D"

1609 ADDITIONAL HOSPITALIZATI SET OF CODES Y:YES
N:NO

M/FORM FID # 56

20 CONVERSION CODE: s cm=Sp(\$p(";Y:YES;N:NO;";";_cm ":";2);";;1),cm=Sp(cm,"";1)

28 STORED: Sp(^ICC(DA,1600),"",8)

29 PREFIELD CODE:

i 'Sd(cbi(55)) s skip="D"
i Sd(cbi(55)) s:cbi(55){ "N"! (cbi(55){ "n") skip="D"

1612 OTHER TREATMENT PROVIDED

M/FORM FID # 15

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone^ptrfilel k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: OTHER TREATMENT PROVIDED (#704)

M/FORM: OTHER TREATMENT PROVIDED (#704)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=^ptc(iform,99)
+e:$d(nocomp) filen=^ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),MDT="" d ^MDT s DT=X k X,Y
d sec^fmfm
d crinck^fmfm
```

II. WINDOWS

OTHER TREATMENT PROVIDED Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 OTHER TREATMENT PROVIDED	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),1605,DA,0),",",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$s('$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle^fmfm d mode0^fmfm
```

1614 DATE OF 1ST EXAM	DATE	S ^DT="E" D ^MDT S X=Y K:Y<1 X
-----------------------	------	--------------------------------

M/FORM FID # 16

20 CONVERSION CODE:

```
s X="(" s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) " " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " $e(cm^"000",9,10) ":" $e(cm^"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X
```

28 STORED: \$p(^ICC(DA,1600),",",11)

1615 TREATMENT DATES	DATE
----------------------	------

M/FORM FID # 17

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(chi(0))>9 s shhfile=1 d fileone`ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$e('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: TREATMENT DATES (#706)

M/FORM: TREATMENT DATES (#706)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=`ptc(ifcom,99)
+s:$d(nocomp) filen=`ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),WOT="" d `WOT s DT=Y k X,Y
d sec`fmfm
d crindx`fmfm
```

II. WINDOWS

TREATMENT DATES Window I

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 TREATMENT DATES	DATE	S WOT="E" D `WOT S X=Y K:Y<1 X

M/FORM FID # 1

20 CONVERSION CODE:

```
s X=" " s:+$e(cm,4,5) X=X $p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".
+$e(cm,4,5)) " " s:+$e(cm,6,7) X=X $e(cm,6,7) ", " s X=X (1700+$e(cm,1,3))
s:$p(cm,".",2)'="" X=X " at " $e(cm`000",9,10) ":" $e(cm`000",11,12) s X=X "_"
s:X="(1700)" X="" s cm=X
```

28 STORED: \$p(`IOC(DA(1),1607,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing=$e('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0`fmfm
```

1616 DATE DISCHARGED FROM TRE	DATE	S WOT="E" D `WOT S X=Y K:Y<1 X
-------------------------------	------	--------------------------------

M/FORM FID # 18

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,1600),"",12)

CA-20 Window 2

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
1713 TOTAL DISABILITY FROM	DATE	S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 33

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,1707),"",1)

1713.1 TOTAL DISABILITY TO	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
----------------------------	------	-------------------------------

M/FORM FID # 34

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,1707),"",2)

44 POST ID:

i 'Sd(cbi(33)) s skip="D"
s X=cm k:XSp(cbi(33),cb,2) X i 'Sd(X) s cm="" s:cm["" cm=""

1714 PARTIAL DISABILITY FROM	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
------------------------------	------	-------------------------------

M/FORM FID # 35

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("IOC(DA,1707),"",3)

1714.1 PARTIAL DISABILITY TO	DATE	S NOT="E" D NOT S X=Y K:Y<1 X
------------------------------	------	-------------------------------

M/FORM FID # 35

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) _:" Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1707),"",4)

44 POST ID:

i 'Sd(abi(36)) s skip="D"
s X=cm k:X\$Sp(abi(36),ab,2) X i 'Sd(X) s cm="" s:cm(""" cm=""

1617 DATE RESUME LIGHT WORK DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 21

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) _:" Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1600),"",13)

1618 DATE RESUME REG WORK DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 22

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) _:" Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1600),"",14)

715 ADVISED EMP CAN RESUME W SET OF CODES Y: YES
N: NO

M/FORM FID # 57

20 CONVERSION CODE: s cm=Sp(Sp(";Y: YES;N: NO;";" _cm _":",2);";",1),cm=Sp(cm,";",1)
28 STORED: Sp(^IOC(DA,1700),"",11)

15.1 DATE OF ADVICE DATE S %DT="E" D %DT S X=Y K:Y<1 X

M/FORM FID # 24

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'=" X=X " at " Se(cm"000",9,10) _:" Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X=" s cm=X

28 STORED: Sp(^IOC(DA,1700),"",12)

29 PREFIELD CODE:

I 'Sd(cbi(57)) s skip="D"
i Sd(cbi(57)) s:cbi(57)["n"!cbi(57)["N"] skip="D"

1619 EXPLAIN EMP WORK LIMITAT

M/FORM FID # 25

28 STORED: nowhere

29 PREFIELD CODE:

s:'Sd(lvl) lvl=0 i Sd(cbi(0))>9 s shhfile=1 d fileone ptrfile1 k shhfile
i Sd(file),file,New=0 s filing=Sd('Sd(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr fmfmm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO-CA f i=lvl:-1:1 s:Sd(DA(i-1)) DA(i)=CA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s:Sd(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: WORK LIMITATIONS (#707)

M/FORM: WORK LIMITATIONS (#707)

I. PRE-FORM CODE

n filen,rptfile,New s:'Sd(nocomp) filen=ptc(iform,99)
+s:Sd(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i 'Sd(DT) s X=Sd(SH),WDT="" d "WDT s DT=Y k X,Y
d sec fmfmm
d crindx fmfmm

II. WINDOWS

WORK LIMITATIONS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 EXPLAIN EMP WORK LIMITAT	WORD PROCESSING	

M/FORM FID # 1

28 STORED: Sp('IOC(DA(1),1608,DA,0),"",1)

III. POST-FORM CODE

i Sd(file),file s filing=Sd('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle fmfmm d node0 fmfmm

2017 ANY PERMANENT EFFECTS	SET OF CODES	Y: YES N: NO

M/FORM FID # 58

20 CONVERSION CODE: s am=Sp(\$p(" ;Y: YES;N: NO; " , " ; " _am_ " : " , 2) , " ; " , 1) , am=Sp(am , " : " , 1)
 28 STORED: Sp(^ICC(DA , 2006) , " " , 5)

54 REMARKS

M/FORM FID # 27

28 STORED: nowhere

29 PREFIELD CODE:

i ' \$d(cbi(58)) s skip= " D "
 i \$d(cbi(58)) s: cbi(58) (" n " i(cbi(58) (" N ") skip= " D "
 s: ' \$d(lvl) lvl= 0 i \$d(cbi(0)) > 9 s shhfile= 1 d fileone^ptrfile1 k shhfile
 i \$d(file) , file , New= 0 s filing= \$s(' \$d(filing) : lvl+ 1 , lvl ' < filing : lvl+ 1 , 1 : filing)
 +d upcntr^fmfm s New= 0 , cbi(0) = " " k file
 s lvl= lvl+ 1
 i lvl s DA(0) = CA , DO= CA f i= lvl : - 1 : 1 s: \$d(DA(i- 1)) DA(i) = CA(i- 1)

30 POSTFIELD CODE:

i lvl s DA(0) = CA f i= 1 : 1 : lvl s: \$d(DA(i)) DA(i- 1) = CA(i)
 s : - 1 DA= DA(0) k DA(lvl) , DA(0) , DI s lvl= lvl- 1

16 CALLS M/FORM: REMARKS (#606)

M/FORM: REMARKS (#606)

I. PRE-FORM CODE

n file1 , rptfile , New s: ' \$d(nocomp) file1 = ^ptc(iform , 99)
 +s: \$d(nocomp) file1 = ^ptg(903 , iform , 99) s New= 0 , rptfile= 1
 s U= " " i ' \$d(DT) s X= \$ad(SH) , WDT= d ^WDT s DT= Y k X , Y
 d sec^fmfm
 d crindx^fmfm

II. WINDOWS

REMARKS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 REMARKS	WORD PROCESSING	

M/FORM FID # 1

28 STORED: Sp(^ICC(DA(1) , 8 , DA , 0) , " " , 1)

III. POST-FORM CODE

i \$d(file) , file s filing= \$s(' \$d(filing) : lvl , lvl > filing : lvl , 1 : filing)
 +d shuffle^fmfm d node0^fmfm

401 REFERRAL PHYSICIAN	FREE TEXT	K: \$L(X) > 30 : (\$L(X) < 1) X
------------------------	-----------	---------------------------------------

M/FORM FID # 37

28 STORED: Sp('ICC(DA,401),'',1)
44 POST ID: s %<cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm('' cm=""

401.1 RP ADDRESS FREE TEXT Y:SL(X)>54!(SL(X)<1) X

M FORM FID # 38

28 STORED: Sp('ICC(DA,401),'',2)
44 POST ID: s %<cm K:SL(X)>54!(SL(X)<1) X s:'Sd(X) cm="" s:cm('' cm=""

401.2 RP CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M FORM FID # 39

28 STORED: Sp('ICC(DA,401),'',3)
44 POST ID: s %<cm K:SL(X)>20!(SL(X)<1) X s:'Sd(X) cm="" s:cm('' cm=""

401.3 RP STATE FORMER 1003' DIE(1003,

M FORM FID # 40

20 CONVERSION CODE: s cm=Ss(cm=""',('Sd('DIE(1003,cm,0))#10):cm,1:Sp('DIE(1003,cm,0),'',1))
28 STORED: Sp('ICC(DA,401),'',4)

401.4 RP ZIPCODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M FORM FID # 41

28 STORED: Sp('ICC(DA,401),'',5)
44 POST ID: s %<cm K:SL(X)>10!(SL(X)<5) X s:'Sd(X) cm="" s:cm('' cm=""

401.5 RP SPECIALTY FREE TEXT K:SL(X)>30!(SL(X)<1) X

M FORM FID # 42

28 STORED: Sp('ICC(DA,401),'',8)
44 POST ID: s %<cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm('' cm=""

2010 REASON FOR REFERRAL SET OF CODES C:CONSULTATION
T:TREATMENT
c:CONSULTATION
t:TREATMENT

M FORM FID # 30

20 CONVERSION CODE: s cm=Sp(9n";C:CONSULTATION;T:TREATMENT;c:CONSULTATION;
t:TREATMENT;";";_cm";",2);";",1).cm=Sp(cm,"",1)
28 STORED: Sp('ICC(DA,2006),'',6)

400.3 DATE SIGNED DATE S NOT="E" D NOT S X=Y K:Y<1 X

M FORM FID # 51

20 CONVERSION CODE:

s X="(" s:+Se(am,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",".",
 ,+Se(am,4,5)) " " s:+Se(am,6,7) X=X Se(am,6,7) " " s X=X (1700+Se(am,1,3))
 s:Sp(am,".",2) X=X " at " Se(am,"000",9,10) ":" Se(am,"000",11,12) s X=X ")"
 s:X="(1700)" X="" s am=X

28 STORED: Sp("ICC(DA,400)", "", 10)

400 ATTENDING PHYSICIAN FREE TEXT K:SL(X)>30!(SL(X)<1) X

M FORM FID # 43

28 STORED: Sp("ICC(DA,400)", "", 1)

44 POST ID: s X=am K:SL(X)>30!(SL(X)<1) X s:'Sd(X) am="" s:am["" am=""

400.1 DOCS ADDRESS FREE TEXT K:SL(X)>54!(SL(X)<1) X

M FORM FID # 44

28 STORED: Sp("ICC(DA,400)", "", 2)

44 POST ID: s X=am K:SL(X)>54!(SL(X)<1) X s:'Sd(X) am="" s:am["" am=""

400.2 DOCS CITY FREE TEXT K:SL(X)>20!(SL(X)<1) X

M FORM FID # 45

28 STORED: Sp("ICC(DA,400)", "", 3)

44 POST ID: s X=am K:SL(X)>20!(SL(X)<1) X s:'Sd(X) am="" s:am["" am=""

400.3 DOCS STATE NUMBER 1003' DIZ/1003.

M FORM FID # 46

20 COMPRESSION CODE: s am=Se(am:"", "", '(Sd("DIZ(1003,am,0))#10):am,1:Sp("DIZ(1003,am,0)", "", 1))

28 STORED: Sp("ICC(DA,400)", "", 4)

400.4 DOCS ZIPCODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M FORM FID # 47

28 STORED: Sp("ICC(DA,400)", "", 5)

44 POST ID: s X=am K:SL(X)>10!(SL(X)<5) X s:'Sd(X) am="" s:am["" am=""

400.8 TAX ID FREE TEXT K:SL(X)>10!(SL(X)<1) X

M FORM FID # 48

28 STORED: Sp("ICC(DA,400)", "", 9)

44 POST ID: s X=am K:SL(X)>10!(SL(X)<1) X s:'Sd(X) am="" s:am["" am=""

400.5 DOES DOCTOR SPECIALIZE OPT OF CODES YES
 NO

M FORM FID # 49

20 CONVERSION CODE: s cm=Sp(;"Y:YES;N:NO;";"_cm ":"",2);";",1),cm=Sp(cm,";",1)
 28 STORED: Sp("IOC(DA,400),"",7)

400.7 SPECIALITY

FREE TEXT

K:SL(X)>30!(SL(X)<1) X

M/FORM FID # 50

28 STORED: Sp("IOC(DA,400),"",8)

29 PREFIELD CODE:

I 'Sd(cbi(49)) s skip="D"

s:cbi(49){"n"!(cbi(49){"N") skip="D"

44 POST ID: s X=cm K:SL(X)>30!(SL(X)<1) X s:'Sd(X) cm="" s:cm("" cm=""

III. POST-FORM CODE

i Sd(file),file,New=0 s:'Sd(lvl) lvl=0
 +s filing=6s('Sd(filing):lvl,lvl>filing:lvl,1:filing)
 +d upcntr="fmcm
 i file=1!(Sd(EPFLAG)) d T3CFILE

CA-20A ATTENDING PHYSICIAN'S SUPPLEMENTAL REPORT

1. NAME OF INJURED EMPLOYEE _____
2. OWCP FILE NUMBER _____
3. HOME MAILING ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
- OVERSEAS _____
4. SOCIAL SECURITY NUMBER _____
5. DATE AND HOUR OF INJURY _____
6. DATE WAGES LOST FROM _____ THROUGH _____
7. DATE OF MOST RECENT EXAMINATION _____
8. INJURED IN LINE OF DUTY? _____
9. IS EMPLOYEE TOTALLY DISABLED FOR WORK? _____
10. *Press <Return> for PRESENT IMPAIRMENT*
11. *Press <Return> for STATE DIAGNOSIS*
12. *Press <Return> for TREATMENT AND FREQUENCY*
13. *Press <Return> for PERMANENT EFFECTS*
14. *Press <Return> for CONCURRENT DISABILITY*
15. DISABILITY CONTINUE FOR 90 OR LONGER? _____
- IF NO, EXPECTED DATE RETURN WORK _____
16. ADVISED EMPLOYEE CAN RESUME WORK? _____ DATE OF ADVICE _____
17. WHEN AND WHAT CAN PARTIALLY DO? _____
- *Press <Return> for DATES AND RESTRICTIONS*
18. REFERRAL PHYSICIAN _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
19. *Press <Return> for PROGNOSIS*
20. ATTENDING PHYSICIAN _____
- ADDRESS _____
- CITY _____ STATE _____ ZIP CODE _____
21. SPECIALITY _____ 22. DATE SIGNED _____

*Word Processing Window will appear.

M/FORM: ON-2DA (#711)

I. PRE-FORM CODE

n filen,rptfile,filing s:'\$d(nocomp) filen='ptc(ifcom,99)
+s:\$d(nocomp) filen='ptg(903,ifcom,99) s New=0,rptfile=1,idedist=2 k ident
s U="" i '\$d(DT) s X=\$d(SH),DT="" d 'DT s DT=Y k X,Y
d sec'fncm

II. WINDOWS

DATA Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
2 NAME	POINTER 1004' EMPLOY(
M/FORM FID # 40		
17 EXTERNAL DEFAULT: NAME		
18 INTERNAL DEFAULT: NAMEPTR		
20 CONVERSION CODE: s cm=6s(cm="":",',(\$d('EMPLOY(cm,0))*10):cm,1:Sp('EMPLOY(cm,0),"",1))		
28 STORED: Sp('IOC(DA,0),"",3)		
32 POST FILING CODE: d setindx'fncm		
44 POST ID: i \$d(new) s:'\$d('IOC("C",new,x1)) msg="Claim not logged",y=24,cm=""		
54 INDEX: 'IOC("C")		
55 LOOKUP CODE:		
40'35'Sp('IOC(Lids,0),"",3)		
9'23'Sp('IOC(Lids,0),"",2)		
137 GACP FILE NUMBER	FREE TEXT	K:SL(X)>131(SL(X)<1) X
M/FORM FID # 2		
28 STORED: Sp('IOC(DA,130),"",6)		
44 POST ID: s X=cm K:SL(X)>131(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""		
7 STREET ADDRESS	FREE TEXT	K:SL(X)>641(SL(X)<1) X
M/FORM FID # 3		
28 STORED: Sp('IOC(DA,1),"",1)		
44 POST ID: s X=cm K:SL(X)>641(SL(X)<1) X s:'\$d(X) cm="" s:cm[""] cm=""		
8 CITY	FREE TEXT	K:SL(X)>231(SL(X)<1)'(X7.AP) X
M/FORM FID # 4		
28 STORED: Sp('IOC(DA,1),"",2)		
44 POST ID: s X=cm K:SL(X)>231(SL(X)<1)'(X7.AP) X s:'\$d(X) cm="" s:cm[""] cm=""		
9 STATE	POINTER 1003' DIZ(1003,	
M/FORM FID # 5		

20 CONVERSION CODE: s cm=Se(cm="":":",('\$(DIZ(1003,cm,0))#10):cm,1:Sp(DIZ(1003,cm,0),"",1))
 28 STORED: Sp(IOC(DA,1),"",3)

10.5 ZIP CODE FREE TEXT K:SL(X)>10!(SL(X)<5) X

M/FORM FID # 6

28 STORED: Sp(IOC(DA,1),"",6)
 44 POST ID: s X=cm K:SL(X)>10!(SL(X)<5) X s:'\$d(X) cm="" s:cm("" Jm=""

9.5 OVERSEAS ADDRESS FREE TEXT K:SL(X)>64!(SL(X)<1) X

M/FORM FID # 7

28 STORED: Sp(IOC(DA,1),"",4)
 44 POST ID: s X=cm K:SL(X)>64!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

3 SSN FREE TEXT K:SL(X)>9!(SL(X)<9)!'(XQ9N) X

M/FORM FID # 8

28 STORED: Sp(IOC(DA,0),"",4)
 44 POST ID: s X=cm K:SL(X)>9!(SL(X)<9)!'(XQ9N) X s:'\$d(X) cm="" s:cm("" cm=""

1 INJURY/ILLNESS DATE & TI DATE S NOT="ET" D "NOT S X=Y K:Y<1 X I SD(X) K:SP(X,".",1)>DT X

M/FORM FID # 9

20 CONVERSION CODE:
 s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp(IOC(DA,0),"",2)
 32 POST FILING CODE: d setindx"fmcm
 54 INDEX: IOC("D")
 55 LOOKUP NODE:
 9"23"Sp(IOC(Lide,0),"",2)
 40"35"Sp(IOC(Lide,0),"",3)

701 DATE WAGES LOST FROM DATE S NOT="E" D "NOT S X=Y K:Y<1 X

M/FORM FID # 41

20 CONVERSION CODE:
 s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",",",
 ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) ", " s X=X (1700+Se(cm,1,3))
 s:Sp(cm,".",2)'="" X=X " at " Se(cm"000",9,10) ":" Se(cm"000",11,12) s X=X ")"
 s:X="(1700)" X="" s cm=X

28 STORED: Sp(IOC(DA,700),"",7)

701.5 DATES WAGES LOST THRU DATE S NOT="E" D "NOT S X=Y K:Y<1 X

M/FORM FID # 11

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,700),"",9)

44 POST ID:

i 'Sd,cbi(41)) s skip="D"
s X=cm k:XSp(cbi(41),cb,2) X i 'Sd(X) s cm="" s:cm(""" cm=""

2003 DATE OF MOST RECENT EXAM DATE S NOT="E" D NOT S Y-Y R:Yd X

M/FORM FID # 12

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm"000",9,10) _:" _Se(cm"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp("ICC(DA,2006),"",1)

307 GROUPED IN LINE OF DUTY SET OF CODES Y:YES
N:NO

M/FORM FID # 13

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO",";" _cm ":" ,2)," ;",1),cm=Sp(cm," :",1)

28 STORED: Sp("ICC(DA,300),"",8)

2004 EMP TOTALLY DISABLED FOR SET OF CODES Y:YES
N:NO

M/FORM FID # 42

20 CONVERSION CODE: s cm=Sp(Sp("Y:YES;N:NO",";" _cm ":" ,2)," ;",1),cm=Sp(cm," :",1)

28 STORED: Sp("ICC(DA,2006),"",2)

2005 DESCRIBE PRESENT INJURY

M/FORM FID # 15

28 STORED: nowhere

29 PREFIELD CODE:

s:'Sd(lvl) lvl=0 i Sd(cbi(0))>9 s attfile=l d filecm=ptrfilel k attfile
i Sd(file),file,New=0 s filing=Se('Sd(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
<d upcntr'filing s New=0,cbi(0)'="" k file
s lvl=lvl+1
i lvl s DA(0)-CA,DO-CA f l=lvl:-1:1 s:Sd(DA(i-1)) DA(i)-CA(i-1)

30 POSTFIELD CODE:

```
i lvl s DA(0)-DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: PRESENT IMPAIRMENT (#712)

M/FORM: PRESENT IMPAIRMENT (#712)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen='ptc(ifcom,99)
+s:$d(nocomp) filen='ptg(903,ifcom,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d(SH),NOT="" d 'NOT s DT=Y k X,Y
d sec'fmfm
d crindx'fmfm
```

II. WINDOWS

PRESENT IMPAIRMENT Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 DESCRIBE PRESENT IMPAIRM	WORD PROCESSING	
M/FORM FID # 1		
28 STORED: \$p('ICC(DA(1),2007,DA,0),"",1)		

III. POST-FORM CODE

```
i $d(file),file s filing=6s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle'fmfm d node0'fmfm
```

2006 20A DIAGNOSIS

M/FORM FID # 16

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(chi(0))>9 s shhfile=1 d fileone'ptrfile1 k shhfile
i $d(file),file,New=0 s filing=6s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upctr'fmfm s New=0,chi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-DA,DO-DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)-DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: SINCE DIAGNOSIS (#713)

M/FORM: SINCE DIAGNOSIS (#713)

I. PRE-FORM CODE

```

n filen,rptfile,New s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d "%DT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm

```

II. WINDOWS

STAGE DIAGNOSIS Window 1

DATA ELEMENT	DATA TYPE	FOUNDER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 20A DIAGNOSIS	WORD PROCESSING	
-------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),2008,DA(0),"",1)

III. POST-FORM CODE

```

i $d(file),file s filing=$e('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm

```

2007 TREATMENT FREQUENCY OF I

M/FORM FID # 17

28 STORED: nowhere

29 POSTFIELD CODE:

```

s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone`ptrfilel k shhfile
i $d(file),file,New=0 s filing=$e('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcnr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)

```

30 POSTFIELD CODE:

```

i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DL s lvl=lvl-1

```

16 CALLS M/FORM: TREATMENT AND FREQUENCY (#714)

M/FORM: TREATMENT AND FREQUENCY (#714)

I. PRE-FORM CODE

```

n filen,rptfile,New s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$d($H),%DT="" d "%DT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm

```

II. WINDOWS

TREATMENT AND FREQUENCY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 TREATMENT/FREQUENCY OF I	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),2009,DA,0),"",1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm
```

CADA Window 2

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
1613 PERMANENT EFFECTS		

M/FORM FID # 18

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $^(obi(0))>9 s shhfile=l d fileone^ptrfile:l k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl<filing:lvl+1,1:filing)
+d upcntr^fmfm s New=0,obi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-CA,D0-CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-CA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)-CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-CA(i)
s:lvl DA-OF(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: PERMANENT EFFECTS (#705)

M/FORM: PERMANENT EFFECTS (#705)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=^ptc(iform,99)
-a:$d(nocomp) filen=^ptq(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad(SH),NOT="" d ^NOT s UT=Y k X,Y
d ser^fmfm
d crindx^fmfm
```

II. WINDOWS

PERMANENT EFFECTS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 PERMANENT EFFECTS	WORD PROCESSING	
-----------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),1606,DA,0),'',1)

III. POST-FORM CODE

```
i $d(file),file s filing-$s('$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle`fmfm d node0`fmfm
```

2209 SUPER STATEMENT AT RECUR

M/FORM FID # 19

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=l d fileone`ptrfilel k shhfile
i $d(file),file,New=0 s filing-$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr`fmfm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)-DA,DO-CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)-DA(i-1)
```

30 POSTFIELD CODE:

```
i lvl s DA(0)-CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)-DA(i)
s:lvl DA-CA(0) k DA(lvl),DA(0),DL s lvl=lvl-1
```

16 CALLS M/FORM: CONCURRENT DISABILITY (#715)

M/FORM: CONCURRENT DISABILITY (#715)

I. PRE-FORM CODE

```
n filen,rptfile,New s:$d(nocomp) filen=`ptc(iform,99)
+s:$d(nocomp) filen=`ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X-$d(SH),WID="" d `NOT s DT=Y k X,Y
d sec`fmfm
d crinck`fmfm
```

II. WINDOWS

CONCURRENT DISABILITY Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 SUPER STATEMENT AT RECUR	WORD PROCESSING	
------------------------------	-----------------	--

M/FORM FID # 1

28 STORED: \$p('ICC(DA(1),211,DA,0),'',1)

III. POST-FORM CODE

i \$d(file),file s filing-\$c('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

2010 WILL DISABIL GO EAST 90 SET OF CODES Y:YES
N:NO

M/FORM FID # 43

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"",":",2),":",1),cm=\$p(cm,":",1)
28 STORED: \$p("ICC(DA,2006)",":",3)

2010.1 EXPECTED DATE RETURN WORK DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 21

20 CONVERSION CODE:

s X="(" s:=\$c(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",":",
,\$c(cm,4,5)) " " s:=\$c(cm,6,7) X=X \$c(cm,6,7) " " s X=X (1700+\$c(cm,1,3))
s:\$p(cm,":",2)'=" X=X " at " \$c(cm,"000",9,10)'=" \$c(cm,"000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: \$p("ICC(DA,2006)",":",4)

29 PREFIELD CODE:

I '\$d(cbi(43)) s skip="D"
i \$d(cbi(43)) s:cbi(43){"Y":(cbi(43){"Y"} skip="D"

44 POST ID:

i '\$d(cbi(43)) s skip="D"
s X=cm k:X(\$p(cbi(43),cb,2) X i '\$d(X) cm="" s:cm["" cm=""

1715 ADVISED EMP CAN RESUME W SET OF CODES Y:YES
N:NO

M/FORM FID # 44

20 CONVERSION CODE: s cm=\$p(\$p("Y:YES;N:NO;"",":",2),":",1),cm=\$p(cm,":",1)
28 STORED: \$p("ICC(DA,1700)",":",11)

1715.1 DATE OF ADVICE DATE S NOT="E" D NOT S X=Y K:Y<1 X

M/FORM FID # 23

20 CONVERSION CODE:

s X="(" s:=\$c(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC",":",
,\$c(cm,4,5)) " " s:=\$c(cm,6,7) X=X \$c(cm,6,7) " " s X=X (1700+\$c(cm,1,3))
s:\$p(cm,":",2)'=" X=X " at " \$c(cm,"000",9,10)'=" \$c(cm,"000",11,12) s X=X "
s:X="(1700)" X=" s cm=X

28 STORED: \$p("ICC(DA,1700)",":",12)

29 PREFIELD CODE:

I '\$d(cbi(44)) s skip="D"

i \$d(cbi(44)) s:cbi(44){ "N"!(cbi(44)){ "n" } skip="D"

2011 WHEN/WHAT CAN EMP PARTIA

M/FORM FID # 24

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s shhfile=1 d fileone~ptrfilal k shhfile
i \$d(file),file,New'=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr~fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,D0=CA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=CA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1

16 CALLS M/FORM: DATES AND RESTRICTIONS (#716)

M/FORM: DATES AND RESTRICTIONS (#716)

I. PRE-FORM CODE

n filen,rptfile,New s:\$d(nocomp) filen=ptc(iform,99)
+s:\$d(nocomp) filen=ptg(903,iform,99) s:New=0,rptfile=1
s D="" i '\$d(DT) s %\$ad(\$H),%DT="" d %DT s DT=X k X,Y
d sec~fufm
d crinck~fufm

II. WINDOWS

DATES AND RESTRICTIONS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

01 WHEN/WHAT CAN EMP PARTIA	WORD PROCESSING	
-----------------------------	-----------------	--

M/FORM FID # 1

28 STORED: Sp('IOC(DA(1),2011,DA(0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing=\$s('\$d(filing):lvl,lvl'<filing:lvl,1:filing)
+d shuffle~fufm d node0~fufm

001 REFERRAL PHYSICIAN	FREE TEXT	K:SL(X)>301(SL(X)<1) X
------------------------	-----------	------------------------

M/FORM FID # 35

28 STORED: Sp('IOC(DA,401),"",1)

44 POST ID: s %cm K:SL(X)>301(SL(X)<1) X s:\$d(X) cm="" s:cm('"' cm=""

401.1 RP ADDRESS

FREE TEXT

K:SL(X)>54!(\$L(X)<1) X

M/FORM FID # 36

28 STORED: \$p('ICC(DA,401),'',2)

44 POST ID: s X=cm K:SL(X)>45!(\$L(X)<1) X s:'\$d(X) cm="" s:cm('' cm=""

401.2 RP CITY

FREE TEXT

K:\$L(X)>20!(\$L(X)<1) X

M/FORM FID # 37

28 STORED: \$p('ICC(DA,401),'',3)

44 POST ID: s X=cm K:SL(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm('' cm=""

401.3 RP STATE

FOUNDER 1003' DIZ(1003,

M/FORM FID # 38

20 CONVERSION CODE: s cm=\$s(cm='':'',('\$d('DIZ(1003,cm,0))#10):cm,1:\$p('DIZ(1003,cm,0),'',1))

28 STORED: \$p('ICC(DA,401),'',4)

401.4 RP ZIPCODE

FREE TEXT

K:SL(X)>10!(\$L(X)<5) X

M/FORM FID # 39

28 STORED: \$p('ICC(DA,401),'',5)

44 POST ID: s X=cm K:SL(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm('' cm=""

2013.20A PROGNOSIS

M/FORM FID # 26

28 STORED: nowhere

29 PREFIELD CODE:

s:'\$d(lvl) lvl=0 i \$d(cbi(0))>9 s sthfile=1 d filecm=ptrfilel k sthfile
i \$d(file),file,New=0 s filing=\$s('\$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr'filing s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO-CA f i=lvl:-1:1 s:\$d(DA(i-1)) DA(i)=CA(i-1)

30 POSTFIELD CODE:

i lvl s DA(0)=CA f i=1:1:lvl s:\$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0).dl s lvl=lvl-1

16 CALLS M/FORM: PROGNOSIS (#717)

M/FORM: PROGNOSIS (#717)

1. PRE-FORM CODE

n filem,rptfile,New s:'\$d(nocomp) filem=ptc(iform,99)
s:\$d(nocomp) filem=ptc(903,iform,99) s New=0,rptfile=1
s '="" i '\$d(UT) s A=\$d(SH),NOT="" d 'NOT s UT=Y k X,Y
d sec'filing

d crinck^fmfm

II. WINDOWS

PROGNOSIS Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 20A PROGNOSIS	WORD PROCESSING	

M/FORM FID # 1

28 STORED: \$p(^ICC(DA,1),2000,DA,0),"",1)

III. POST-FORM CODE

i \$d(file),file s filing-\$s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d shuffle^fmfm d node0^fmfm

400 ATTENDING PHYSICIAN	FREE TEXT	K:\$L(X)>30!(\$L(X)<1) X
-------------------------	-----------	--------------------------

M/FORM FID # 28

28 STORED: \$p(^ICC(DA,400),"",1)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

400.1 DOCS ADDRESS	FREE TEXT	K:\$L(X)>54!(\$L(X)<1) X
--------------------	-----------	--------------------------

M/FORM FID # 29

28 STORED: \$p(^ICC(DA,400),"",2)

44 POST ID: s X=cm K:\$L(X)>45!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

400.2 DOCS CITY	FREE TEXT	K:\$L(X)>20!(\$L(X)<1) X
-----------------	-----------	--------------------------

M/FORM FID # 30

28 STORED: \$p(^ICC(DA,400),"",3)

44 POST ID: s X=cm K:\$L(X)>20!(\$L(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

400.3 DOCS STATE	POINTER 1003'	DIZ(1003,
------------------	---------------	-----------

M/FORM FID # 31

20 CONVERSION CODE: s cm=\$s(cm="":'(\$d(DIZ(1003,cm,0))#10):cm,1:\$p(DIZ(1003,cm,0),"",1))

28 STORED: \$p(^ICC(DA,400),"",4)

400.4 DOCS ZIPCODE	FREE TEXT	K:\$L(X)>10!(\$L(X)<5) X
--------------------	-----------	--------------------------

M/FORM FID # 32

28 STORED: \$p(^ICC(DA,400),"",5)

44 POST ID: s X=cm K:\$L(X)>10!(\$L(X)<5) X s:'\$d(X) cm="" s:cm("" cm=""

400.7 SPECIALITY

FREE TEXT

K:\$L(X)>30!(\$L(X)<1) X

M/FORM FID # 34

28 STORED: Sp(^IOC(DA,400),"",8)

44 POST ID: s X=cm K:\$L(X)>30!(\$L(X)<1) X s:'\$d(X) cm="" s:cm(" " cm=""

400.9 DATE SIGNED

DATE

S \$DT="E" D ^DT S X=Y K:Y<1 X

M/FORM FID # 33

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." ,+\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at "\$e(cm"000",9,10)" ":"\$e(cm"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^IOC(DA,400),"",10)

III. POST-FORM CODE

i \$d(file),file,New=0 s:'\$d(lvl) lvl=0
+s filing=6s('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fncfm
i file=1!(\$d(ERFLAG)) d ^T3OFILE

M/FORM: TICKLER (#727)

I. PRE-FORM CODE

```
n filen,rptfile,New,filing,DA s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),%DT="" d ^%DT s DT=Y k X,Y
d sec^fmfm
```

II. WINDOWS

TICKLER FILE Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
150 TICKLER FILE		

M/FORM FID # 2

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(obi(0))>9 s shhfile=1 d fileone^ptrfilel k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upontr^fmfm s New=0,obi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=DA,DO=DA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=DA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=DA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=DA(i)
s:lvl DA=DA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: TICKLER FILE (#728)

M/FORM: TICKLER FILE (#728)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen=ptc(iform,99)
+s:$d(nocomp) filen=ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),%DT="" d ^%DT s DT=Y k X,Y
d sec^fmfm
d crindk^fmfm
```

II. WINDOWS

TICKLER FILE Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
.01 TICKLER FILE	NUMERIC	K:X'~X1(X0999999)!(X<1)!(X7.E1". "IN.N) X

M/FORM FID # 1

28 STORED: \$p(^ICC(DA(1),150,DA,0),"",1)

1 DATE TO DO DATE S WDT="E" D ^WDT S X=Y R:Y<1 X

M/FORM FID # 2

20 CONVERSION CODE:

s X="(" s:+\$e(cm,4,5) X=X \$p("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","."
+,\$e(cm,4,5)) " " s:+\$e(cm,6,7) X=X \$e(cm,6,7) " " s X=X (1700+\$e(cm,1,3))
s:\$p(cm,".",2)'=" X=X " at " \$e(cm,"000",9,10)'=" \$e(cm,"000",11,12) s X=X ")"
s:X="(1700)" X="" s cm=X

28 STORED: \$p(^ICC(DA(1),150,DA,0),"",2)

2 RESPONSIBLE TO DO FREE TEXT K:SL(X)>3!(SL(X)<1) X

M/FORM FID # 3

28 STORED: \$p(^ICC(DA(1),150,DA,0),"",3)

44 POST ID: s X=cm K:SL(X)>3!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

3 ACTION TO TAKE FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 4

28 STORED: \$p(^ICC(DA(1),150,DA,0),"",4)

44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'\$d(X) cm="" s:cm("" cm=""

III. POST-FORM CODE

i \$d(file),file s filing=\$e('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d node0'filing

2 NAME POINTER 1004' EMPLOY(

M/FORM FID # 1

20 CONVERSION CODE: s cm=\$e(cm:"":'('\$d(EMPLOY(cm,0))#10):cm,1:\$p(EMPLOY(cm,0),"",1))

28 STORED: \$p(^ICC(DA,0),"",3)

32 POST FILING CODE: d setindx'filing

54 INDEX: ^ICC("C")

55 LOOKUP NODE: 1^20^\$p(^ICC(Lids,0),"",3)

III. POST-FORM CODE

i \$d(file),file,new=0 s:'\$d(lvl) lvl=0
+s filing=\$e('\$d(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr'filing

M/FORM: TICKLER (#727)

I. PRE-FORM CODE

```
n filen,rptfile,New,filing,DA s:'$d(r,comp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),WDT="" d ^WDT s DT=Y k X,Y
d sec^fufm
```

II. WINDOWS

TICKLER FILE Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

150 TICKLER FILE

M/FORM FID # 2

28 STORED: nowhere

29 PREFIELD CODE:

```
s:'$d(lvl) lvl=0 i $d(cbi(0))>9 s shhfile=1 d fileone^ptrfile1 k shhfile
i $d(file),file,New=0 s filing=$s('$d(filing):lvl+1,lvl'<filing:lvl+1,1:filing)
+d upcntr^fufm s New=0,cbi(0)="" k file
s lvl=lvl+1
i lvl s DA(0)=CA,DO=CA f i=lvl:-1:1 s:$d(DA(i-1)) DA(i)=CA(i-1)
k DA(0) S DA="",DI=""
```

30 POSTFIELD CODE:

```
i lvl s DA(0)=CA f i=1:1:lvl s:$d(DA(i)) DA(i-1)=CA(i)
s:lvl DA=CA(0) k DA(lvl),DA(0),DI s lvl=lvl-1
```

16 CALLS M/FORM: TICKLER FILE (#728)

M/FORM: TICKLER FILE (#728)

I. PRE-FORM CODE

```
n filen,rptfile,New s:'$d(nocomp) filen='ptc(iform,99)
+s:$d(nocomp) filen='ptg(903,iform,99) s New=0,rptfile=1
s U="" i '$d(DT) s X=$ad($H),WDT="" d ^WDT s DT=Y k X,Y
d sec^fufm
d crink^fufm
```

II. WINDOWS

TICKLER FILE Window 1

DATA ELEMENT	DATA TYPE	POINTER GLOBAL REFERENCE OR INPUT VERIFICATION
--------------	-----------	--

.01 TICKLER FILE

NUMERIC

K:~X'~X1(X0999999)1(X1)1(X2.E1".IN.N) X

M/FORM FID # 1

28 STORED: Sp(^IOC(DA(1),150,DA,0),"",1)

1 DATE TO DO DATE _WDT="E" D ^WDT S X=Y K:Y<1 X

M/FORM FID # 2

20 CONVERSION CODE:

s X="(" s:+Se(cm,4,5) X=X Sp("JAN.FEB.MAR.APR.MAY.JUN.JUL.AUG.SEP.OCT.NOV.DEC","." +Se(cm,4,5)) " " s:+Se(cm,6,7) X=X Se(cm,6,7) " " s X=X (1700+Se(cm,1,3))
s:Sp(cm,".",2)'="" X=X " at " _Se(cm^"000",9,10) _:" _Se(cm^"000",11,12) s X=X _)"
s:X="(1700)" X="" s cm=X

28 STORED: Sp(^IOC(DA(1),150,DA,0),"",2)

2 RESPONSIBLE TO DO FREE TEXT K:SL(X)>3!(SL(X)<1) X

M/FORM FID # 3

28 STORED: Sp(^IOC(DA(1),150,DA,0),"",3)

44 POST ID: s X=cm K:SL(X)>3!(SL(X)<1) X s:'Sd(X) cm="" s:cm(""" cm=""

3 ACTION TO TAKE FREE TEXT K:SL(X)>60!(SL(X)<1) X

M/FORM FID # 4

28 STORED: Sp(^IOC(DA(1),150,DA,0),"",4)

44 POST ID: s X=cm K:SL(X)>60!(SL(X)<1) X s:'Sd(X) cm="" s:cm(""" cm=""

III. POST-FORM CODE

i Sd(file),file s filing=Se('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d node0^fmfm

2 NAME FORMIER 1004* EMPLOY(

M/FORM FID # 1

20 CONVERSION CODE: s cm=Se(cm=""":'',"('Sd(^EMPLOY(cm,0))#10):cm,1:Sp(^EMPLOY(cm,0),"",1))

28 STORED: Sp(^IOC(DA,0),"",3)

32 POST FILING CODE: d setindx^fmfm

54 INDEX: ^IOC("C")

55 LOOKUP NODE: 1^20^Sp(^IOC(lids,0),"",3)

III. POST-FORM CODE

i Sd(file),file,New=0 s:'Sd(lvl) lvl=0
+s filing=Se('Sd(filing):lvl,lvl>filing:lvl,1:filing)
+d upcntr^fmfm

APPENDIX B
FASCIMILE OF CLAIM FORMS

**APPENDIX B
FASCIMILE OF CLAIM FORMS**

FORM LABEL	NAME OF FORM	M/FORM ID #
CA-1	Federal Employee's Notice of Traumatic Injury and Claim for Continuation of Pay/Compensation	613
CA-2	Notice of Occupational Disease and Claim for Compensation	644
CA-2A	Notice of Employee's Recurrence of Disability and Claim for Pay/Compensation	651
CA-3	Report of Termination of Disability and or Payment	657
CA-5	Claim for Compensation by Widow, and/or Children	672
CA-5B	Claim for Compensation by Parents, Brothers, Sisters, Grandparents, or Grandchildren	688
CA-6	Official Superior's Report of Employee's Death	658
CA-7	Claim for Compensation on Account of Traumatic Injury or Occupational Disease	671
CA-8	Claim for Continuing Compensation on Account of Disability	662
CA-16	Authorization for Examination and/or Treatment	687
CA-17	Duty Status Report	689
CA-20	Attending Physician's Report	708
CA-20A	Attending Physician's Supplemental Report	711

CA-1 Federal Employee's Notice of Traumatic Injury and Claim for Continuation of Pay/Compensation

Employee: Please complete all boxes 1 - 15 below. Do not complete shaded areas.

Witness: Complete bottom section 16.

Employing Agency: (Supervisor or Compensation Specialist): Complete shaded boxes a, b, and c.

1. Name of employee (Last, First, Middle)

2. Social Security Number

3. Date of birth

4. Sex

5. Home telephone

6. Grade as of injury
Level Step

7. Employee's home mailing address

8. Dependents

9. Place where injury occurred (e.g. 2nd floor, Main Post Office Bldg., 12th & Pine)

10. Date injury occurred Time

11. Date of this notice

12. Employee's occupation

13. Cause of injury (Describe what happened and why)

14. Nature of injury (identify both the injury and the part of body, e.g., fracture of left leg)

a. Occupation Code

b. Type code

c. Source code

d. OWCP Use-NOI Code

15. I certify, under penalty of law, that the injury described above was sustained in performance of duty as an employee of the United States Government and that it was not caused by my willful misconduct, intent to injure myself or another person, nor by my intoxication. I hereby claim medical treatment, if needed, and the following, as checked below, while disabled for work:

Signature of employee or person acting on his/her behalf

Any person who knowingly makes any false statement, misrepresentation, concealment of fact, or any other act of fraud to obtain compensation as provided by the FECA or who knowingly accepts compensation to which that person is not entitled, is subject to felony criminal prosecution and may, under appropriate provisions, be punished by a fine or imprisonment, or both.

Have your supervisor complete the receipt attached to this form and return it to you for your records.

End of Employee Report

16. Statement of witness (Describe what you saw, heard, or know about this injury)

Name of witness

Signature of witness

Date signed

ADDRESS (Include City, State, Zip code)

CA-1 REVISION DATE: 1/86

CA-1 Federal Employee's Notice of Traumatic Injury and Claim for Continuation of Pay/Compensation
Official Supervisor's Report: Please complete information requested below:

17. Agency name and address of reporting office

OWCP Agency Code
OSHA Site Code

18. Employee's duty station

19. Regular work hours (From - To)

20. Regular work schedule

21. Date of Injury

22. Date notice received

23. Date stopped work

24. Date pay stopped

25. Date 45 day period began

26. Date returned to work

27. Was employee injured in performance of duty?

28. Was injury caused by employee's willful misconduct, intoxication, or intent to injure self or another?

29. Was injury caused by third party?

30. Name and address of third party

31. Name and address of physician first providing medical care

32. First date medical care received

33. Do medical reports show employee is disabled for work?

34. Does your knowledge of the facts about this injury agree with statements of the employee and/or witness?

35. Does the employing agency controvert continuation of pay?

36. Pay rate when employee stopped work

37. A supervisor who knowingly certifies to any false statement, misrepresentation, concealment of fact, etc., in respect to this claim may also be subject to appropriate felony criminal prosecution.

I certify that the information given above and that furnished by the employee on the reverse of this form is true to the best of my knowledge with the following exception:

Name of supervisor

Signature of supervisor

Date

Supervisor's Title

Office Phone

38. Filing instructions

CA-1 REVISION DATE: 3. 96

OFFICE OF OCCUPATIONAL DISEASE AND CLAIMS FOR COMPENSATION

Employee: Please complete all boxes 1 - 13 below. Do not complete shaded areas.

Employing Agency (Supervisor or Compensation Specialist): Complete shaded boxes a, b, and c.

-
- | | | | | |
|---|--------|-------------------|---|--|
| 1. Name of employee (Last, First, Middle) | | | 2. Social Security Number | |
| 3. Date of birth | 4. Sex | 5. Home telephone | 6. Grade as of date of last exposure
Level Step | |
| 7. Employee's home mailing address | | | 8. Dependents | |
| 9. Employee's occupation | | | a. Occupation Code | |
| 10. Location (address) where you worked when disease or illness occurred | | | 11. Date 1st aware | |
| 12. Date you first realized the disease or illness was caused or aggravated by your employment | | | | |
| 13. Explain the relationship to your employment, and why you came to this realization | | | | |
| 14. Nature of disease or illness | | | a. OWP Use-NOI Code
b. Type code c. Source code | |
| 15. If this claim was not filed with the employing agency within 30 days after date shown above in item #12, explain the reason for the delay | | | | |
| 16. If the statement requested in item 1 of the attached instructions is not submitted with this form, explain reason for delay. | | | | |
| 17. If the medical reports requested in item 2 of attached instructions are not submitted with this form, explain reason for delay. | | | | |
| 18. I certify, under penalty of law, that the disease or illness described above was the result of my employment with the United States Government, and that it was not caused by my willful misconduct, intent to injure myself or another person, nor by my intoxication. I hereby claim medical treatment, if needed, and other benefits provided by the Federal Employee's Compensation Act | | | | |

Signature of employee or person acting on his/her behalf

Date

Have your supervisor complete the receipt attached to this form and return it to you for your records. Any person who knowingly makes any false statement, misrepresentation, concealment of fact, or any other act of fraud to obtain compensation as provided by the FECA or who knowingly accepts compensation to which that person is not entitled, is subject to felony criminal prosecution and may, under appropriate provisions, be punished by a fine or imprisonment, or both.

OA-2 REVISION DATE: 3.86

Agency Code

19. Agency name, and address of reporting office

OSHA Site Code

20. Employee's duty station

22. Regular work schedule

23. Name and address of physician first providing medical care

24. First date medical care received

25. Do medical reports show employee is disabled for work?

26. Date employee first reported condition to supervisor

27. Date and hour employee stopped work

28. Date and hour employee's pay stopped

29. Date employee was last exposed to conditions alleged to have caused disease or illness

30. Date returned to work

31. If employee has returned to work and work assignment has changed, describe new duties

32. Was injury caused by third party?

33. Name and address of third party

34. A supervisor who knowingly certifies to any false statement, misrepresentation, or concealment of fact, etc., in respect to this claim may also be subject to appropriate felony criminal prosecution.

I certify that the information given above and that furnished by the employee on the reverse side of this form is true to the best of my knowledge with the following exception:

Name of Supervisor

Signature of Supervisor

Date

Supervisor's Title

Office phone

OSHA REVISION DATE: 3/85

CA-2A Notice Of Employee's Recurrence Of Disability And Claim For Pay/Compensation
IMPORTANT: Before completing this form please read carefully the instructions.

-
1. NAME OF INSURED EMPLOYEE 2. SOCIAL SECURITY NUMBER 3. OWCP File number for original injury (if known)
4. HOME MAILING ADDRESS 5. HOME TELEPHONE
6. NAME AND ADDRESS OF EMPLOYING ESTABLISHMENT at time of original injury 7. NAME AND ADDRESS OF EMPLOYING ESTABLISHMENT at time of recurrence, if other than 6.
8. DATE AND HOUR of original injury 9. DATE AND HOUR of recurrence 10. DATE AND HOUR stopped work following recurrence 11. DATE AND HOUR pay stopped following recurrence
12. PAY RATE IN EFFECT ON: Base Pay Subsistence Quarters Other Pay
- a. Date of Recurrence (\$ per)
- b. Date Stopped Work Following Recurrence (\$ per)
13. Show work week at time pay stopped, if other than Monday thru Friday 14. DATE AND HOUR returned to work, following recurrence
15. At time of recurrence did official superior authorize medical treatment?
16. DATE employee first received medical
17. NAME AND ADDRESS of physician treating employee following recurrence treatment.
18. After returning to work following the original injury, was the employee handicapped or in any way limited in performing his/her usual duties?

19. Describe the circumstances of the recurrence of disability as reported by the employee. If the condition gradually worsened over a period of time, describe the progress of the condition from the time employee returned to work up to the date of recurrence.

A supervisor who knowingly certifies to any false statement, misrepresentation, concealment of fact, etc., in respect to this claim may also be subject to appropriate felony criminal prosecution.

20. Signature of official superior (at time of recurrence) 21. Title 22. Official superior's work phone number 23. Date

CA-2A REVISION DATE: 9/86

CA-2A Notice Of Employee's Recurrence Of Disability And Claim For Pay/Compensation

24. Inclusive dates that employee's regular pay continued during this period of recurrence. Do not include period of sick or annual leave. From: Thru:
25. Show gross dollar amount of regular pay which employee received during this period of recurrence.
26. If pay changed during the period employee continuation of was receiving continuation of pay, for this recurrence, show date of change. Base Subistence Quarters Other (Specify)
27. If pay rate changed during the period employee was receiving pay, give new rate.
28. Complete this item if you worked during the period shown in item 29(b) or 29(c).
a. Dates & Hours Worked b. Pay Rate c. Total Amount Earned d. Type Work Performed e. Name & Address of Employer
29. I certify that the recurrence claimed on date in item 9 was due to the injury shown in item 8 and I hereby claim medical treatment, if needed, and the following as checked below, while disabled for work.
- a. Sick and/or annual leave
From: Thru:
- b. Continuation of regular pay not to exceed 45 days, which will include days taken during the original injury and prior recurrence(s), and compensation for wage loss if disability for work continues beyond 45 days. (If my claim is denied, I understand that the continuation of my regular pay shall be charged to sick or annual leave, or be deemed an overpayment within the meaning of 5 USC 5584).
From: Thru:
- c. Continuing compensation on account of occupational disease.
From: Thru:
30. Signature of Employee or Person Acting on his/her behalf. Any person who knowingly makes any false statement, misrepresentation, concealment of fact, or any other act of fraud to obtain compensation as provided by the FECA or who knowingly accepts compensation to which that person is not entitled is subject to felony criminal prosecution and may, under appropriate criminal provisions, be punished by a fine or imprisonment, or both.
31. Date

CA-2A REVISION DATE: 9/86

1. Name of Injured Employee (Last, first, middle)
2. Social Security Number
3. OWCP File Number (if known)
4. Department or Agency
5. Bureau or Agency
6. Name and Address of Reporting Office
7. Date and Hour of Injury
8. Date and Hour Stopped Work
9. Date and Hour Pay Stopped
10. Date and Hour Returned to Work
11. Employee's Work Week On Return To Duty If Other Than Monday Through Friday
12. Present Pay Rate If Different From That Received At Time Employee Stopped Work
 - a. Base Pay
 - b. Subsistence
 - c. Quarters
 - d. Other (Specify)
13. Inclusive Dates Employee Received Pay For Any Part of The Period of Absence Because of:
 - a. Annual Leave
 - From:
 - Through:
 - b. Sick Leave
 - From:
 - Through:
 - c. Other (Specify)
 - From:
 - Through:
14. Has Employee's Work Assignment Been Changed Because of Disability Resulting From This Injury?
15. If Interrupted, Show Dates Deductions For Health Benefits and/or Optional Insurance Were Resumed
 - Health Benefit
 - Optional Insurance
16. If Health Benefits Option Has Changed Since Disability Began, Show New Code Number and Date of Change
 - Number
 - Date
17. Remarks:
18. Inclusive Dates That The Employee's Regular Pay Continued During The Period of Disability.
 - Do not include period of sick or annual leave.
 - From:
 - Through:
19. Show The Gross Dollar Amount Of Regular Pay Which The Employee Received During The Period of Disability.
 - Do not include pay received for sick or annual leave.
20. If Pay Rate Changed During The Period Employee Was Receiving Continuation of Pay, Show The Date of Change.
21. If Pay Rate Changed During The Period Employee Was Receiving Continuation of Pay, Give New Rate
 - a. Base Pay
 - b. Subsistence
 - c. Quarters
 - d. Other (Specify)
22. Signature of Supervisor
23. Title and Office Phone Number
24. Date

CA-3 REVISION DATE:

-
1. Name of deceased employee 2. Date of Birth 3. Date of Injury 4. Date of Death 5. Social Security Number
6. Name and address of employing agency 7. Nature of injury which caused death
8. Name and address (Include Zip Code) 9. Your Date of Birth 10. Date of Marriage to Employee
11. Were you living with the employee at time of death? 12. Were you ever married to anyone other than the employee? 13. Was employee ever married to anyone other than yourself?
14. List all employee's children from this marriage who may be entitled to compensation (See attached information sheet for definition of children):
- | Name | Relationship | Date of Birth | Address |
|------|--------------|---------------|---------|
|------|--------------|---------------|---------|
15. If a legal guardian has been appointed for any child named above, give name of child, name and address of the guardian.
- | Child | Guardian | Guardian's Address |
|-------|----------|--------------------|
|-------|----------|--------------------|
16. List other relatives who were fully or partially dependent on employee:
- | Name | Relationship | Date of Birth | Address |
|------|--------------|---------------|---------|
|------|--------------|---------------|---------|
17. If employee was ever in the Armed Forces of the United State, give: Service number:
Branch of service:
Period of service:
18. If application has been made for Veteran's Administration (VA) benefits because of employee's death, give:
VA Claim number: Address of VA office where claim is filed:
19. If application has been made for U.S. Civil Service Annuity because of employee's death give:
CSF Claim Number: Date Annuity began: Amount paid per month \$
20. If a claim has been made against a third party because of employee's death, give: Amount of recovery:
Name and address of third party.
21. Total burial expense 22. Amount of burial expense paid or payable by VA.
23. Name and address of party (other than VA) whose funds were used to pay burial expense and amount paid:
- I hereby certify that each and every statement made above is true to the best of my knowledge.
24. Signature of person filing claim 25. Address (Include Zip code) 26. Date

CR-5 REVISION DATE: 12-86

CA-5 Claim for Compensation by Widow, Widower, and/or Children
Attending Physician's Report

-
1. Name of deceased employee (Last, first, middle)
 3. What history of injury or employment related disease was given to you?
 4. If treated for disease, give diagnosis.
 5. If death was not instantaneous, describe the treatment you provided.
 6. Show date on which treatment was given:
 7. What was the direct cause of death?
 8. What were the contributory causes of death, if any?
 9. In your opinion, was the death of the employee due to the injury as reported in item 3 above?
Give the medical reasons for your opinion, unless causal relationship is obvious.
 10. Was a biopsy or an autopsy performed?
If yes, give name and address of physician and arrange for a copy of the report to be submitted.
 11. Name and address (please type - include Zip code)
 12. Signature
 13. Date signed
-

CA-5 REVISION DATE: 12/86

1. Name of deceased employee	2. Date of Birth	3. Date of Injury
4. Date of Death		5. Social Security Number
6. Name and address of employing agency		7. Nature of injury which caused death
8. Name of dependent (Last, first, middle)		9. Dependent's address
10. Dependent's birth date		11. Dependent's Occupation
12. Dependent's Social Security Number	13. Dependent's relationship to employee	14. Extent of dependency on employee
15. Total amount employee contributed to dependent's support during 12 months immediately prior to death?	16. Did employee live with dependent during the 12 months immediately prior to death? if "yes", complete 17 & 18	17. Total amount employee paid dependent in money or service for room and board in addition to the amount shown in 15. \$ per
		18. If no fixed amount was paid what is the fair value of of such? \$ per
19. If dependent was employed during 12 month period prior to employee's death, give: Type of work performed: Period of employment: Monthly pay rate: Name and address of employer:		20. Show dependent's income from all sources other than employment during 12 month period prior to employee's death: Investment \$ Pensions Person other than employee Other Total
1. Birth Date	22. Occupation	23. Monthly pay rate \$
		24. Total income from all source for 12 months prior to employee's death \$
25. List all property owned by dependent and husband or wife (omit clothing, furniture, personal item) Description	Date Acquired	Value
26. If employee was ever in the Armed Forces of the United States, give: Service number: Branch of service: Period of service:		27. If an application has been made for Veteran's Administration (VA) benefits because of employee's death, give: VA Claim number: Address of VA office where claim is filed:
28. If an application has been made for U.S. Civil Service Annuity because of employee's death, give: CSF Claim Number: Date Annuity began: Amount paid per month: \$		29. If a claim has been made against a third party because of employee's death, give: Amount of recovery: \$ Name and address of third party:
30. Total burial expense \$	31. Amount of burial expense paid or payable by VA \$	32. Name and address of party (other than VA) whose funds were used to pay burial expense and amount paid: \$
33. Signature of person filing claim	34. Address	35. Date

Attending Physician's Report

-
1. Name of deceased employee (Last, first, middle) 2. Date of death
3. What history of injury or employment related disease was given to you? 4. If treated for disease, give diagnosis.
5. If death was not instantaneous, describe the treatment you provided. 6. Show date on which treatment was given.
7. What was the direct cause of death?
8. What were the contributory causes of death, if any?
9. In your opinion, was the death of the employee due to the injury as reported in Item 3 above?
10. Was a biopsy or an autopsy performed? (Arrange for a copy of the report to be submitted.)
11. Name and address
12. Signature 13. Date Signed (Mo., day, year)
-

CA-5B REVISION DATE: 9/86

-
1. Name of Deceased Employee (Last, first, middle) 2. Date of Birth 3. SEX 4. Social Security No.
Department or Agency 6. OACP Agency Code 7. OSHA Site Code
8. Name and Address of Reporting Office 9. Name and Office Phone Number of Employee's Official Superior
10. Date and Hour of Injury 11. Date and Hour of Death 12. Date and Hour Employee's Pay Stopped
13. Describe how injury occurred 14. Was employee in performance of duty when injury occurred?
15. Location where injury occurred 16. Location where death occurred
17. Immediate cause of death (Attach medical and autopsy report if available)
18. Employee's pay rate as of a. Base Pay b. Subsistence c. Quarters d. Other
A. Date of Injury \$ per \$ per \$ per \$ per
B. Date pay stopped \$ per \$ per \$ per \$ per
19. Did the employee work in position held at time of injury for a full eleven months immediately prior to the injury?
20. If answer to 19 is no, would position have afforded employment for eleven months except for the injury?
21. Did employee receive leave pay for any part of period from time pay stopped to date of death? (Give inclusive dates) 22. a. Occupation code
b. Type code c. Source code
d. OACP use-NOI coded.
23. Did employee receive continuation of pay (COP) during period prior to death?
a. Pay rate used for COP \$ per
b. Inclusive dates of CO from to
24. If employee was enrolled in Health Benefit Plan for self and family, show HBS Code Number: 25. Show date through which HBS deductions were last made (Mo., day, yr) 26. If employee received medical care prior to death, give name and address of attending physician
27. If injury was caused by third party, give name and address of third party. 28. Give name and address of attorney representing survivors if legal action is instituted against the third party.
29. Show amount of third party recovery, if any.
30. If employee was a member of the Armed Forces of the United States, show:
Branch of Service:
Serial No. (if known) 31. Has claim for survivor's benefits been filed with the Office of Personnel Management?
32. Name and address of employee's spouse or next of kin (Show relationship, if other than spouse)
33. Signature of Official Superior 34. Title 35. Date (Mo., day, year)
-

CA-6 REVISION DATE: 11/86

CA-7 Claim for Compensation On Account of Traumatic Injury or Occupational Disease

1. Name of Employee _____ 2. OWCP File Number _____
3. Social Security Number _____ 4. Period of wage loss for which compensation is claimed
From _____ Thru _____ Hours
schedule _____
5. Is this a claim for a award?
6. Has any pay been received for period shown in item 4? _____ 7. If yes, amount _____ From _____ Thru _____
8. Was claim made against 3rd party? _____ 9. Name and address of 3rd party or insurance carrier _____
10. Has the claim been settled? Give amount recovered. _____
11. Have you ever applied for or received benefits from the Veterans Administration based on disability incurred while serving in the Armed Forces of the United States? If yes, furnish ->
a. Claim number b. Address of VA office where claim is filed c. Nature of disability and monthly payment.
12. Have you applied for or received any annuity under the U.S. Civil Service Retirement Act or any other Federal Retirement or Disability Law? If yes, furnish ->
a. Claim number b. Date annuity began c. Amount of monthly payment.
13. List your dependents
Name _____ Date of Birth _____ Relationship Living with you? Mailing Address, if different from your own _____
14. Support information for above dependents
Are you making support payments for a dependent shown above? _____
15. Were support payments ordered by a court? If so, attach copy of court order.
16. If yes, support payments are made to: _____ 17. Amount _____ Per _____
18. I hereby make claim for compensation because of the injury sustained by me while in the performance of my duty for the United States, said injury not being due to willful misconduct on my part or to my intention to bring about the injury or death of myself or another, or to my intoxication. I have been disabled because of this injury and have not refused or failed to perform any work I was able to do during the period for which compensation is claimed, and every statement above is true to the best of my knowledge and belief.
- Any person who knowingly makes any false statement, misrepresentation, concealment of fact, or any other act of fraud, to obtain compensation under the Federal Employee's Compensation Act, or who knowingly accepts compensation to which that person is not entitled is subject to felony criminal prosecution and may, under appropriate criminal provisions, be punished by a fine or imprisonment, or both.
- Employee's Signature _____ Date _____
19. Employee's home mailing address _____

20. Pay Rate As Of: a. Base Pay b. Subsistence c. Quarters d. Other (Specify)
 \$ per \$ per \$ per \$ per

Date of Injury
 Date Employee
 Stopped Work

21. If employee received additional pay, identify type and show amount

Premium Pay (per) Night Pay (per)
 Sunday Pay (per) Other (Identify)

22. Show work schedule for week pay stopped

23. Did employee work in position for 11 months prior to injury?

24. If not, would position have afforded employment
 for 11 months but for the injury?

25. Total length of federal civilian service
 Yrs. Mos.

26. Was the employee enrolled in a Health Benefits
 Program on the date pay stopped?
 If yes, give code.

27. Was the employee enrolled in an Optional Life
 Insurance Program on the date pay stopped?
 If yes, was employee enrolled in Option A, B, or C?

Ending date of the pay period in which mo day yr
 HBS/OLI Deductions were last made?

If Option B, show number of multiples

28. Type and inclusive dates employee received leave for any part of period since stopping work.

Sick Leave (mo/day/yr) From: Thru:
 Annual Leave (mo/day/yr)
 Other Leave (mo/day/yr)

29. If employee received continuation of pay (COP), give dates.

From: Thru:

30. Date all pay stopped (mo/day/yr hr of day)

31. Period for which compensation is claimed (mo/day/yr)
 From: Thru:

32. Date returned to work (mo/day/yr hr of day)

33. Work schedule when returned to work

34. Did the work assignment change because of disability
 resulting from the injury?

35. Pay rate on return to work
 \$ per

36. A supervisor who knowingly certifies to any false statement, misrepresentation, or concealment of fact,
 with respect to this claim may also be subject to appropriate felony criminal prosecution.

I certify that the information given above and that furnished by the employee on the reverse side of this form
 is true to the best of my knowledge with the following exception:

Signature of Supervisor

Date

Supervisor's title

Agency name & address

Office phone number

37. If OWP needs specific pay information the person who should be contacted is
 Phone Number.

1. NAME OF INJURED EMPLOYEE (Last, first, middle) 2. OACP FILE NUMBER, IF KNOWN
3. HOME MAILING ADDRESS (Include ZIP code) 4. SOCIAL SECURITY NUMBER

5. DATE AND HOUR OF INJURY (Mo., day, year)
6. PERIOD COMPENSATION IS CLAIMED AS RESULT OF PAY LOSS (Mo., day, year). IF PAY LOSS WAS INTERMITTENT ATTACH
SEPARATE SHEET SHOWING DATES AND HOURS OF PAY LOSS. FROM: THROUGH:

7. HAVE YOU RECEIVED ANY LEAVE PAY DURING THE PERIOD SHOWN IN ITEM 6?

8. DO YOU WISH TO REPURCHASE LEAVE?
SICK LEAVE FROM THROUGH:
ANNUAL LEAVE FROM THROUGH:
OTHER LEAVE THROUGH:
OTHER LEAVE SPECIFIED:

9. COMPLETE THIS ITEM IF YOU WORKED DURING THE PERIOD SHOWN IN ITEM 6. ATTACH A SEPARATE SHEET IF NEEDED.
a. SALARIED EMPLOYMENT.

DATES & HOURS WORKED	PAY RATE (Per hour, day or week)	TOTAL AMOUNT EARNED	TYPE WORK PERFORMED	NAME & ADDRESS OF EMPLOYER
-------------------------	-------------------------------------	------------------------	------------------------	-------------------------------

b. COMMISSION AND SELF-EMPLOYMENT. SHOW ALL ACTIVITIES, WHETHER OR NOT INCOME RESULTED FROM YOUR EFFORTS.

DATES & HOURS WORKED	NAME AND ADDRESS OF BUSINESS	SELF-EMPLOYED COMMISSION	TYPE OF ACTIVITY PERFORMED	INCOME DERIVED (ATTACH EXPLANATION IF NEEDED)
-------------------------	---------------------------------	-----------------------------	-------------------------------	--

10. IF YOU HAVE APPLIED FOR EMPLOYMENT WITH THE U.S. TRAINING AND EMPLOYMENT SERVICE GIVE THE FOLLOWING:

REGISTRATION NO.	DATE OF REGISTRATION	OFFICE ADDRESS
------------------	----------------------	----------------

11. IF YOU WERE ONLY PARTIALLY DISABLED AND DID NOT WORK, STATE REASON FOR NOT WORKING.

12. IF, SINCE FILING YOUR INITIAL CLAIM FOR COMPENSATION, YOU HAVE APPLIED FOR OR RECEIVED VA BENEFITS BASED ON
MILITARY SERVICE FOR THE UNITED STATES, GIVE THE FOLLOWING:

CLAIM NO.	NATURE OF DISABILITY AND MONTHLY PAYMENT	NAME AND ADDRESS OF OFFICE WHERE CLAIM IS FILED
-----------	--	--

13. IF, SINCE FILING YOUR INITIAL CLAIM FOR COMPENSATION, YOU HAVE APPLIED FOR OR RECEIVED AN ANNUITY UNDER THE
CIVIL SERVICE RETIREMENT ACT OR OTHER FEDERAL RETIREMENT OR DISABILITY LAW, GIVE THE FOLLOWING:

CLAIM NO.	AMOUNT OF MONTHLY PAYMENT	NAME AND ADDRESS OF OFFICE WHERE CLAIM IS FILED
-----------	---------------------------	--

14. SIGNATURE OF EMPLOYEE OR PERSON ACTING ON EMPLOYEE'S BEHALF. Any person who
knowingly makes any false statement, misrepresentation, concealment of fact or any other act of fraud to
obtain compensation as provided by the FECA or who knowingly accepts compensation to which that per-
son is not entitled is subject to felony criminal prosecution and may, under appropriate criminal provisions,
be punished by a fine or imprisonment, or both.

15. DATE (mo., day, year)

16. IF EMPLOYEE HAS RETURNED TO WORK, SHOW DATE AND HOUR (Mo., day, year) OTHER THAN MONDAY THRU FRIDAY

17. HAS EMPLOYEE RECEIVED ANY PAY FOR WORK, LEAVE, SUBSISTENCE, QUARTERS OR OTHER REMUNERATION FROM YOUR AGENCY DURING THE PERIOD SHOWN IN ITEM 6 ON THE REVERSE SIDE?

19. IF ANSWER TO ITEM 18 IS YES, SHOW:

AMOUNT: \$
TYPE OF PAYMENT:
PERIOD: FROM THROUGH

20. IF THERE HAS BEEN ANY CHANGE IN EMPLOYEE'S HEALTH BENEFIT ENROLLMENT AND/OR OPTIONAL INSURANCE SINCE PREVIOUS CLAIM FOR COMPENSATION WAS SUBMITTED, PLEASE EXPLAIN. (i.e. change of plan or option; if additional deductions have been made by the agency, show amount and period.)

21. REMARKS

22. SIGNATURE OF OFFICIAL SUPERIOR

23. TITLE

24. DATE (mo., day, year)

CA-8 REVISION DATE: 10/86

The following request for information is authorized by law (5 USC 8101 et. seq.). Benefits and/or medical services expenses may not be subject to suspension under this program unless this report is completed and filed as requested. Information collected will be handled in compliance with the Freedom of Information Act, the Privacy Act of 1974 and OMB Cir.No.A-108.

1. Name and Address of the Medical Facility or Physician Authorized to provide the Medical Service:

2. Employee's Name (Last, first, middle)

3. Date of Injury (Mo., day, yr.)

4. Occupation

5. Description of Injury or Disease:

6. You are authorized to provide medical care for the employee for a period of up to 60 days from the date shown in item 11, subject to the condition stated in item A, and to the condition indicated either 1 or 2, in item B.
- A. Your signature in item 35 of Part B certifies your agreement that all fees for services shall not exceed the maximum allowable fee established by OACP and that payment by OACP will be accepted as payment in full for said services.

B. Conditions:

7. If a Disease or Illness is Involved, OACP Approval for Issuing Authorization was Obtained from: (Type Name and Title of OACP Official)

8. Signature of Authorizing Official:

9. Name and Title of Authorizing Official: (Type or print clearly)

10. Local Employing Agency Telephone Number:

11. Date (Mo., day, year)

12. Send one copy of your report: (Fill in remainder of address)

13. Name and Address of Employee's Place of Employment:

U.S. DEPARTMENT OF LABOR
Employment Standards Administration
Office of Worker's Compensation Programs

Department or Agency
Bureau or Office
Local Address

CA-16 REVISION DATE: 10/86

4. Employee's Name (Last, first, middle)

5. What History of Injury or Disease Did Employee Give You?

. Is There Any History or Evidence of Concurrent or Pre-existing Injury, Disease, or Physical Impairment?
(If yes, please describe)

7. What Are Your Findings? (Include results of X rays, laboratory tests, etc.)

8. What Is Your Diagnosis?

9. Do You Believe the Condition Found was Caused or Aggravated by the Employment Activity Described?
(Please explain your answer if there is doubt.)

10. Did Injury Require Hospitalization?
If yes, date of admission (Mo., day, year)
Date of discharge (Mo., day, year)

21. Is Additional Hospitalization Required?

2. Surgery (If any, describe type)

23. Date Surgery Performed (Mo., day, year)

4. What (Other) Type of Treatment Did You Provide?

25. What Permanent Effects, If Any, Do you Anticipate?

6. Date of First Examination (Mo., day, year)

27. Date(s) of Treatment (Mo., day, year)

. Date of Discharge From Treatment (Mo., day, year)

9. Period of Disability (Mo., day, year) (If termination date unknown, so 30.)
Is Employee Able to Resume Indicate.

Light work Date:

Total Disability (From To)

Regular Work Date:

Partial Disability (From To)

1. If Employee is Able to Resume Work, Has He/She Been Advised? If Yes, Furnish Date Advised

2. If Employee is Able to Resume Only Light Work, Indicate The Extent of Physical Limitations and The Type of Work That Could Reasonably be Performed with These Limitations.

3. General Remarks and Recommendations for Future Care, If Indicated. If You Have Made a Referral to Another Physician or to a Medical Facility, Provide Name and Address.

1. Do You Specialize? (If yes, state specialty)

5. SIGNATURE OF PHYSICIAN. I certify that all statements in response to the questions asked in part B of this form are true complete and correct to the best of my knowledge. Further, I understand that any false or misleading statement or any misrepresentation or concealment of material fact which is knowingly made may subject me to felony criminal prosecution.

Address (No., Street, City, State, ZIP Code).

37. Tax Identification Number

38. Date of Report

This request for information is authorized by law (5 USC 8101 et seq.) Benefits and/or medical expenses may not be paid or may be subject to suspension under the Federal Employee's Compensation Program unless this report is completed and filed as requested. Information collected will be handled and stored in compliance with the Freedom of Information Act, the Privacy Act of 1974 and the OMB Cir.A-108.

Instructions for Completing and Submitting this Form

Supervisor: Complete Part A and refer the form to the attending physician for completion of Part B.

Attending Physician: Complete Part B. To prevent interruption of the employee's pay, the completed form should be returned to the employing agency (as shown in Item 12) within two days following examination and/or treatment. A copy of the form should also be sent to the shown in Item 11).

1. Name and Address of Medical Facility Providing Medical Services:

2. OWP File Number (if known)

3. Employee's Name:

4. Date of Injury:

5. Social Security No:

6. Occupation:

7. Describe How the Injury Occurred and State Parts of the Body Affected.

8. Specify the Usual Work Requirements of the Employee. Check Whether Employee Performs These Tasks or is Exposed Continuously or Intermittently, and Give Number of Hours per day.

Activity

Identify or Describe

Type Exposure: Hrs/Day Degrees dBA

9. Does the Job Require Driving a Vehicle? Operating Machinery?

10. The Employee Works: Hours Per Day Day Per Week

11. Send a Copy of This Report To:

U.S. Department of Labor
Employment Standards Administration
Office of Workers' Compensation Programs

12. Send the Original Report to (Name and Address of Employing Agency):

13a. Does the History of Injury Given to You by the Employee correspond to that Show in Item 7?

13b. Description of Clinical Findings

13c. Diagnosis of Condition Due to Injury

13d. Diagnosis of Other Disabling Conditions

14. Is Employee Able to Perform His/Her Regular Work? Full-Time or Part-Time? if Part-Time, hours per day:

15. Complete the following if the answer to Item 14 is "No"

Activity if not, complete Item 15 below	Identify or Describe	Type Exposure	Hrs/Day	Degrees dBA
---	----------------------	---------------	---------	-------------

16. Describe Any Other Function of This Employee's Regular Work Which is Medically Restricted By The Injury.

17. Period of Disability

Total Disability (From-To)

18. If Employee is Able to Resume Work, Has He/She Been Advised?
If yes, Give Date of Advice.

19. Date of Examination

20. Date of Next Appointment, If Scheduled

21. Typed or Printed Name and Address of Physician

22. Specialty

23. Tax Identification Number

24. Physician's Signature

25. Date

CA-17 REVISION DATE: 4/87

1. Patient's name 2. Date of injury 3. O&P File Number
4. Is there any history or evidence of concurrent or pre-existing injury or disease or physical impairment? (If yes, please describe.) ICD-9 CODE
5. What are your findings? (Include results of X-Rays, laboratory reports, etc.)
6. What is your diagnosis? ICD-9 Code
7. Do you believe the condition found was caused or aggravated by the employment activity described?
8. Did injury require hospitalization?
If no, go to item #12 9. Date of admission 10. Date of discharge
11. Additional Hospitalization required? If Yes, describe in Item 24.
12. What treatment did you provide?
13. Date of first examination 14. Date(s) of treatment 15. Date of discharge
from treatment
16. Period of total disability
From Thru 17. Period of Partial Disability
From Thru 18. Date employee able to
resume light work.
19. Date employee is able to resume regular work.
20. Has employee been advised that he/she can
return to work? 21. If yes, on what date was he/she advised?
22. If employee is able to resume only light work, indicate the extent of physical limitations and the type of
work that could reasonably be performed with these limitations. (Continue item #24 if necessary.)
23. Are any permanent effects expected as a result of this injury? If, yes describe in item #24.
24. Remarks.
25. If you have referred the employee to another physician provide the following: Specialty
26. What was the reason for this referral?
27. I certify that the statements in response to the questions asked above are true, complete and correct to the best
of my knowledge. Further, I understand that any false or misleading statement or any misrepresentation or conceal-
ment of material fact which is knowingly made may subject me to felony criminal prosecution.
- Signature of Physician Date
28. Name and address of Physician 29. Tax ID Number
30. Do you specialize? 31. If yes, indicate specialty

CA-20 REVISION DATE: 10/86

FOR INSTRUCTIONS SEE REVERSE SIDE

-
- | | | |
|---|--|--|
| 1. NAME OF INJURED EMPLOYEE | 2. OWCIP FILE NUMBER, IF KNOWN | |
| 3. HOME MAILING ADDRESS (Include ZIP code) | 4. SOCIAL SECURITY NUMBER | |
| 5. DATE AND HOUR OF INJURY | 6. PERIOD COMPENSATION IS CLAIMED AS RESULT OF PAY LOSS
FROM: THROUGH: | |
| 7. DATE OF MOST RECENT
EXAMINATION (Mo., day, year) | 8. IS EMPLOYEE'S PRESENT CONDITION
DUE TO THE INJURY FOR WHICH
COMPENSATION IS CLAIMED? | 9. IS EMPLOYEE TOTALLY DISABLED
FOR USUAL WORK? |
| 10. DESCRIBE NATURE OF PRESENT IMPAIRMENT | 11. STATE DIAGNOSIS | |
| 12. WHAT TREATMENT IS EMPLOYEE RECEIVING AND HOW OFTEN IS IT GIVEN? | | |
| 13. WHAT PERMANENT EFFECTS, IF ANY, ARE ANTICIPATED? | | |
| 14. DESCRIBE ANY CONCURRENT DISABILITY EMPLOYEE HAS WHICH IS NOT RELATED TO THIS INJURY | | |
| 15. WILL DISABILITY FOR REGULAR WORK CONTINUE FOR
90 DAYS OR LONGER?
IF NO, APPROXIMATELY WHAT DATE WILL EMPLOYEE BE
ABLE TO RETURN TO WORK? | 16. IF EMPLOYEE IS ABLE TO RESUME REGULAR WORK, HAS
HE OR SHE BEEN ADVISED?
IF YES, SHOW DATE EMPLOYEE WAS INFORMED | |
| 17. EMPLOYEE IS ONLY PARTIALLY DISABLED, SHOW DATE
HE OR SHE WAS ABLE TO PERFORM SOME WORK AND
DESCRIBE SPECIFIC WORK RESTRICTIONS. (i.e. Limitations in
stooping, bending, lifting, etc.) | 18. IF EMPLOYEE HAS BEEN REFERRED TO ANOTHER
PHYSICIAN FOR CONSULTATION OR TREATMENT, GIVE
PHYSICIAN'S NAME & ADDRESS. | |
| 19. RECOMMENDATIONS AND PROGNOSIS | | |
| 20. ADDRESS (Include ZIP code) | 21. IF YOU SPECIALIZE, INDICATE SPECIALTY | |
| 22. SIGNATURE OF PHYSICIAN. I certify that the statements on the
reverse apply to this report and are made a part hereof. | 23. DATE OF REPORT (Mo., day, year) | |

CA-20A REVISION DATE: 10/86

APPENDIX C
LIST FILE ATTRIBUTES

**APPENDIX C
LIST FILE ATTRIBUTES**

FILE NUMBER	FILE NAME
1002	FORMS
1004.2	INJURY COMP & CLAIM (EMPLOYEE)
1005	INJURY COMPENSATION & CLAIMS
1007	ACTIVITY FILE
1017	NATURE OF INJURY (NOI)
1018	BODY PART
1025	ACCIDENT FILE
1026	CAUSE OF INJURY
1027	SEALANCE OF INJURY
1029	CLAIM TYPE
1057	OSHA INJURY CODE
1061	ADJUDICATION STATUS
1062	OWCP AGENCY CODE
1063	OWCP REPORTING OFFICE
1068	AGENT OF ACCIDENT
1093	PAY STATUS

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
The forms package was created to print CA form facsimiles. Five routines are run in conjunction with this file: T3CF, T3CF1, T3CFB, and, T3CFI.			
'IDENTIFIED' BY TITLE OF FORM			
'POINTED TO' BY THE 'NEXT FORM' FIELD OF THE 'NEXT FORM' SUB-FIELD OF THE 'FORMS' FILE			
'POINTED TO' BY THE 'CLAIM FORM' FIELD OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD OF THE 'EMPLOYEE' FILE			
'POINTED TO' BY THE 'CLAIM FORM NEXT' FIELD OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD OF THE 'EMPLOYEE' FILE			
'POINTED TO' BY THE 'CLAIM FORM LAST RECEIVED' FIELD OF THE 'ICC FILE' FILE			
'POINTED TO' BY THE 'CLAIM FORM NEXT' FIELD OF THE 'ICC FILE' FILE			
CROSS-REFERENCED BY: NAME OF FORM(B), SHORT NAME(C), TITLE OF FORM(D)			
CREATED AUG 27, 1987 (NHFC)			
1002,.01	NAME OF FORM	0;1	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>6!(SL(X)<1)!(X'7LP.E) X CROSS-REFERENCE: 1002'B 1)= S FORMS("B",SE(X,1,30),DA)="" 2)= K FORMS("B",SE(X,1,30),DA)
1002,1	TITLE OF FORM	0;2	FREE TEXT ANSWER MUST BE 1-100 CHARACTERS IN LENGTH Input transform: K:X[*****]!(SA(X)=45) X I SD(X) K:SL(X)>100!(SL(X)<1) X LAST EDITED: OCT 18, 1987 CROSS-REFERENCE: 1002'D 1)= S FORMS("D",SE(X,1,30),DA)="" 2)= K FORMS("D",SE(X,1,30),DA)
1002,2	REVISION	0;3	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH Input transform: K:SL(X)>6!(SL(X)<1) X
1002,3	HEADER	1;0	1002.01 Multiple
1002.01,.01	HEADER	0;1	FREE TEXT ANSWER MUST BE 1-132 CHARACTERS IN LENGTH Input transform: K:SL(X)>132!(SL(X)<1) X LAST EDITED: SEP 29, 1987
1002,3.5	HEADER PAGE #2	5;0	1002.07 Multiple
1002.07,.01	HEADER PAGE #2	0;1	FREE TEXT ANSWER MUST BE 1-132 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>132!(SL(X)<1) X LAST EDITED: SEP 29, 1987
1002,4	INSTRUCTIONS	2;0	1002.02 WORD-PROCESSING
1002,4.5	SECTION	6;0	1002.08 Multiple 'IDENTIFIED' BY SECTION NAME

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1002.08,.01	SECTION POSITION	0;1	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 99 Input transform: K:+X'=X!(X>99)!(X<0)!(X2.E1". "IN.N) X LAST EDITED: SEP 14, 1987 CROSS-REFERENCE: 1002.08"B 1)= S FORMS(DA(1),6,"B",SE(X,1,30),DA)="" 2)= K FORMS(DA(1),6,"B",SE(X,1,30),DA)
1002.08,1	SECTION NAME	0;2	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:X!""""!(SA(X)=45) X I SD(X) K:SL(X)>60!(SL(X)<1) X LAST EDITED: SEP 14, 1987 CROSS-REFERENCE: 1002.08"C 1)= S FORMS(DA(1),6,"C",SE(X,1,60),DA)="" 2)= K FORMS(DA(1),6,"C",SE(X,1,60),DA)
1002,5	DATA ELEMENTS OR ITEMS	3;0	1002.03 Multiple 'IDENTIFIED' BY SHORT NAME 'IDENTIFIED' BY MORE THAN ONE REF
1002.03,.01	NUMBER ON FORM	0;1	FREE TEXT ANSWER MUST BE 1-4 CHARACTERS IN LENGTH (Required) (Multiply asked) Input transform: K:SL(X)>4!(SL(X)<1) X CROSS-REFERENCE: 1002.03"B 1)= S FORMS(DA(1),3,"B",SE(X,1,30),DA)="" 2)= K FORMS(DA(1),3,"B",SE(X,1,30),DA)
1002.03,1	SHORT NAME	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:X!""""!(SA(X)=45) X I SD(X) K:SL(X)>30!(SL(X)<1) X CROSS-REFERENCE: 1002"C 1)= S FORMS("C",SE(X,1,30),DA(1),DA)="" 2)= K FORMS("C",SE(X,1,30),DA(1),DA)
1002.03,2	QUESTION TEXT	1;0	1002.04 Multiple
1002.04,.01	QUESTION TEXT	0;1	FREE TEXT ANSWER MUST BE 1-132 CHARACTERS IN LENGTH (Multiply) Input transform: K:SL(X)>132!(SL(X)<1) X LAST EDITED: SEP 29, 1987
1002.04,1	PRINT COLUMN	0;2	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 160 Input transform: K:+X'=X!(X>160)!(X<1)!(X2.E1". "IN.N) X LAST EDITED: SEP 17, 1987
1002.04,2	PRINT ROW	0;3	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 66 Input transform: K:+X'=X!(X>66)!(X<1)!(X2.E1". "IN.N) X LAST EDITED: SEP 17, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1002.03,3	SECTION ON FORM	0;3	FREE TEXT XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:X[""]!(SA(X)=45) X I SD(X) K:SL(X):50!(SL(X)<1) X I 'SD('FORMS(DA(1),6,"C",X)) W !,"NOT DEFINED!" K X LAST EDITED: SEP 14, 1987 CROSS-REFERENCE: 1002.03"C 1)= S 'FORMS(DA(1),3,"C",SE(X,1,60),DA)="" 2)= K 'FORMS(DA(1),3,"C",SE(X,1,60),DA)
1002.03,4	FILE CROSS REFERENCE	2;0	1002.05 Multiple
1002.05,.01	FILE CROSS REFERENCE	0;1	NUMBER TYPE A NUMBER BETWEEN .01 AND 99999.999 (Multiply asked) Input transform: K:X=X*(X<99999.999)!(X<.01)!(X>.E1"."4N.N) X LAST EDITED: OCT 30, 1987 CROSS-REFERENCE: 1002.03"AB 1)= S 'FORMS(DA(2),3,"AB",SE(X,1,30),DA(1),DA)="" 2)= K 'FORMS(DA(2),3,"AB",SE(X,1,30),DA(1),DA)
1002.05,.05	SUB-FIELD	0;4	NUMBER XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER ENTER A SUB-FILE NUMBER OF WORKING FILE Input transform: K:X=X*(X<9999999999.99)!(X<0)!(X>.E1"."6N.N) X I SD(X) K:'SD('TD(X)) X I SD(X) K:X="!"(X'(\$P('FORMS(D0,0),U,4)) X NEW I,B,E S B=\$P('FORMS(DA,0),U,4),B-B+.9999999999 I B'=" F I=1:1 S B=\$O('TD(B)) Q:B>E W !,B,715,\$P('TD(B,0),U,1) LAST EDITED: SEP 18, 1987 CROSS-REFERENCE: 1002.05"AC"LMPE 1)= I X'=" I SD('TD(X,0,"UP")) S UP='TD(X,0,"UP") I SD('TD(UP,"SB",X)) S SP=\$O('TD(UP,"SB",X,"")) SP=\$P(\$P('TD(UP,SP,0),U,4),";",1) SP('FORMS(D0,3,D1,2,D2,0),U,5)-SP 2)= S \$P('FORMS(D0,3,D1,2,D2,0),U,5)=""
1002.05,1	PRINT COLUMN	0;2	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 160 Input transform: K:X=X*(X<132)!(X<1)!(X>.E1"."1N.N) X LAST EDITED: SEP 17, 1987
1002.05,2	PRINT ROW	0;3	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 66 Input transform: K:X=X*(X<66)!(X<1)!(X>.E1"."1N.N) X LAST EDITED: SEP 17, 1987
1002.05,3	SUB-FIELD NODE	0;5	FREE TEXT ANSWER MUST BE 1-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<1) X WRITE AUTHORITY: ^ LAST EDITED: SEP 18, 1987
1002.05,4	POINTER FILE FIELD	0;6	SET ; '1' FOR NAME FIELD ; '2' FOR SECOND FIELD LAST EDITED: OCT 13, 1987
1002.03,5	MORE THAN ONE REF	0;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: JAN 21, 1988

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1002.03,6	PAGE NUMBER	0;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 9 (Required) Input transform: K:+X'~X1(X09)!(X<1)!(X2.E1)". "IN.N) X LAST EDITED: OCT 2, 1987
1002,6	NOTES/GENERAL INFORMATION 4;0		1002.06 WORD-PROCESSING
1002,7	NOHIMS/OSHRMS FILE	0;4	POINTER TO FILE FILE (#1) (Required) LAST EDITED: SEP 17, 1987
1002,8	MFORM NUMBER	0;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 500 AND 999999 Input transform: K:+X'~X1(X099999)!(X<500)!(X2.E1)". "IN.N) X LAST EDITED: SEP 23, 1987
1002.9	NEXT FORM	7;0	1002.09POINTER Multiple
1002.09,.01	NEXT FORM	0;1	POINTER TO FORMS FILE (#1002) (Multiply asked) LAST EDITED: SEP 29, 1987 CROSS-REFERENCE: 1002.09`B 1)= S `FORMS(DA(1),7,"B",SE(X,1,30),DA)="" 2)= K `FORMS(DA(1),7,"B",SE(X,1,30),DA)

FILES POINTED TO

FIELDS

FILE (#1)	NOHIMS/OSHRMS FILE (#7)
FORMS (#1002)	NEXT FORM:NEXT FORM (#.01)

INPUT TEMPLATE(S):

COL/ROW	NOV 2,1987 USER #0
INPUTS BY FORM, NUMBER ON FORM, TEXT, TEXT COL, TEXT ROW, CROSS REF, CROSS REF COL, CROSS REF ROW	
DONNA'S REFS	OCT 30,1987 USER #0
INPUTS BY FORM, NUMBER ON FORM, CROSS REF, CROSS REF SUB-FIELD(NODE), CROSS REF POINTER, MORE THAN ONE REF, POINTERS	NOV 3,1987 USER #0

PRINT TEMPLATE(S):

CAPTIONED	USER #0	
CROSS REF	OCT 28,1987 USER #0	FORMS LIST
D2	OCT 30,1987 USER #0	FORMS/EMPLOYEE REFERENCES
LIST FORMS	NOV 10,1987 USER #0	FORMS LIST

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHIMS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
'IDENTIFIED' BY INJURY/ILLNESS DATE & TIME CROSS-REFERENCED BY: CLAIM FORM(B), INJURY/ILLNESS DATE & TIME(C), TYPE OF PAY RECEIVED(P1), DATE IMPLEMENTED(P2)			
1004.2,.001	CLAIMS FORM ENTRY NUMBER		NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 999999999 Input transform: K:~X'~X!(X>999999999)!(X<1)!(X?.E1)."IN.N) X LAST EDITED: SEP 21, 1987
1004.2,.01	CLAIM FORM	0;1	POINTER TO FORMS FILE (#1002) (Required) Input transform: S DIC("S")="I \$D(DIA("P")) I SP("(0),U,4)=DIA("P")" D DIC K DIC S DIO=DIE,X~>Y K:Y<0 X SELECT ONE OF THE FOLLOWING CLAIM FORMS LAST EDITED: FEB 3, 1988 CROSS-REFERENCE: 1004.2"B 1)= S EMPLOY(DA(1),"ICC","B",SE(X,1,30),DA)="" 2)= K EMPLOY(DA(1),"ICC","B",SE(X,1,30),DA)
1004.2,1	INJURY/ILLNESS DATE & TIME	0;2	DATE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER (Required) Input transform: S WDT="ET" D WDT S X~Y K:Y<1 X I \$D(X) K:SP(X,".",1)>DT X LAST EDITED: NOV 2, 1987 CROSS-REFERENCE: 1004.2"C"MMPS 1)= S EMPLOY(DA(1),"ICC","C",SP(X,".",1),DA)="" 2)= K EMPLOY(DA(1),"ICC","C",SP(X,".",1),DA)
1004.2,1.7	INJURY LOG ENTRY DATE	0;9	DATE Input transform: S WDT="E" D WDT S X~Y K:Y<1 X
1004.2,3	SSN	0;4	FREE TEXT ANSWER MUST BE 9 CHARACTERS IN LENGTH Input transform: K:SL(X)>9!(SL(X)<9)!(X?N) X
1004.2,4	SEX	0;5	SET ; 'F' FOR FEMALE ; 'M' FOR MALE ; 'U' FOR UNKNOWN
1004.2,5	BADGE NUMBER	0;6	FREE TEXT ANSWER MUST BE 3-11 CHARACTERS IN LENGTH Input transform: K:SL(X)>11!(SL(X)<3) X
1004.2,6	BIRTHDATE	0;7	DATE Input transform: S WDT="E" D WDT S X~Y K:Y<1 X
1004.2,7	STREET ADDRESS	1;1	FREE TEXT ANSWER MUST BE 1-64 CHARACTERS IN LENGTH Input transform: K:SL(X)>64!(SL(X)<1) X LAST EDITED: FEB 3, 1988
1004.2,8	CITY	1;2	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:SL(X)>23!(SL(X)<1)!(X?.AP) X LAST EDITED: AUG 28, 1987
1004.2,9	SURVE	1;3	POINTER TO SURVE FILE (#1003)
1004.2,9.5	OVERSEAS ADDRESS	1;4	FREE TEXT ANSWER MUST BE 1-64 CHARACTERS IN LENGTH Input transform: K:SL(X)>64!(SL(X)<1) X

DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/NOIDS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,10	HOME PHONE NUMBER	1;5	FREE TEXT FORMAT: (AREA CODE) XXX-XXXX Input transform: R:SL(X)>151(SL(X)<1) X
1004.2,10.5	ZIP CODE	1;6	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: R:SL(X)>101(SL(X)<5) X LAST EDITED: AUG 14, 1987
1004.2,11	AGENCY	2;1	POINTER TO AGENCY UNIT FILE (#1074) LAST EDITED: AUG 20, 1987
1004.2,11.5	AGENCY UNIT	2;2	POINTER TO AGENCY UNIT FILE (#1074) LAST EDITED: AUG 20, 1987
1004.2,12	SHIP	2;3	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: R:SL(X)>301(SL(X)<1) X
1004.2,13	DUTY STATION	9;1	POINTER TO OWP AGENCY CODES FILE (#1062) LAST EDITED: SEP 15, 1987
1004.2,13.1	OWP AGENCY CODE	9;2	FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH Input transform: R:SL(X)>61(SL(X)<6) X LAST EDITED: SEP 15, 1987
1004.2,13.2	ACTIVITY UIC	9;3	FREE TEXT ANSWER MUST BE 5-6 CHARACTERS IN LENGTH Input transform: R:SL(X)>61(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,13.3	CLAIMANT CODE	9;4	FREE TEXT ANSWER MUST BE 2-4 CHARACTERS IN LENGTH Input transform: R:SL(X)>41(SL(X)<2) X LAST EDITED: SEP 15, 1987
1004.2,13.4	DUTY STATION ADDRESS	9;5	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: R:SL(X)>541(SL(X)<1) X LAST EDITED: SEP 17, 1987
1004.2,13.5	DUTY STATION CITY	9;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: R:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,13.6	DUTY STATION STATE	9;7	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,13.7	DUTY STATION ZIP CODE	9;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: R:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,13.9	OSHA SITE CODE	9;10	FREE TEXT ANSWER MUST BE 1-9 CHARACTERS IN LENGTH Input transform: R:SL(X)>91(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14	REPORTING OFFICE	10;1	POINTER TO OWP REPORTING OFFICE FILE (#1063) LAST EDITED: SEP 15, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSIRIS/ACHSIS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,14.01	PO ADDRESS	10;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 17, 1987
1004.2,14.02	PO CITY	10;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.03	PO STATE	10;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,14.04	PO ZIP CODE	10;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,14.05	POINT OF CONTACT	10;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.055	POC TITLE	10;10	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.06	POC PHONE	10;7	FREE TEXT ANSWER MUST BE 10-13 CHARACTERS IN LENGTH Input transform: K:SL(X)>13!(SL(X)<10) X LAST EDITED: SEP 15, 1987
1004.2,14.07	POC EXTENSION	10;8	FREE TEXT ANSWER MUST BE 1-4 CHARACTERS IN LENGTH Input transform: K:SL(X)>4!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.08	POC AUTOMOB	10;9	FREE TEXT ANSWER MUST BE 7-13 CHARACTERS IN LENGTH Input transform: K:SL(X)>13!(SL(X)<7) X LAST EDITED: SEP 15, 1987
1004.2,14.5	EMPLOYEE COST CENTER	2;4	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH Input transform: K:SL(X)>6!(SL(X)<1) X
1004.2,15	OCCUPATION	3;1	POINTER TO OCCUPATION FILE (#1001) (Required) LAST EDITED: SEP 15, 1987
1004.2,15.5	OCCUPATION CODE	3;2	FREE TEXT ANSWER MUST BE 5-6 CHARACTERS IN LENGTH Input transform: K:SL(X)>6!(SL(X)<5) X LAST EDITED: AUG 28, 1987
1004.2,16	TENURE	3;3	SET ; 'TEMP' FOR TEMPORARY ; 'NON' FOR NONCOMPETING ; 'CARE' FOR CAREER ; 'CARC' FOR CAREER-CONDITIONAL.
1004.2,17	PAY PLAN	3;4	FREE TEXT EXAMPLES: (GS, WG, WD, GM) Input transform: K:SL(X)>2!(SL(X)<2)!(X22A) X
1004.2,18	PAY GRADE	3;5	FREE TEXT FORMAT: (01, 12, 09, 13) Input transform: K:SL(X)>3!(SL(X)<2) X

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,19	PAY STEP	3;6	FREE TEXT ENTER THE LEVEL 01-99. Input transform: K:SL(X)>21:(SL(X)<1) X
1004.2,20	PAY BASIS	3;7	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 1, 1987
1004.2,20.2	PAY INFORMATION	5;0	1004.21 SET (Add New Entry without Asking) Multiple 'IDENTIFIED' BY DATE IMPLEMENTED
1004.21,.01	TYPE OF PAY RECEIVED 0;1		SET (Required) ; '1' FOR REGULAR/BASE PAY ; '2' FOR CONTINUATION OF PAY ; '3' FOR COMPENSATION PAY ; '4' FOR ANNUAL LEAVE PAY ; '5' FOR SICK LEAVE PAY ; '6' FOR OTHER LEAVE PAY ; '7' FOR MISCELLANEOUS PAY LAST EDITED: OCT 18, 1987 CROSS-REFERENCE: 1004.21,1 1)= S 'EMPLOY(DA(2),"ICC","P1",SE(X,1,30),DA(1),DA)="" 2)= K 'EMPLOY(DA(2),"ICC","P1",SE(X,1,30),DA(1),DA)
1004.21,1	DATE IMPLEMENTED	0;2	DATE Input transform: S 'NOT="E" D "NOT S X=Y K:Y< LAST EDITED: OCT 18, 1987 CROSS-REFERENCE: 1004.21,2 1)= S 'EMPLOY(DA(2),"ICC","P2",SE(X,1,30),DA(1),DA)="" 2)= K 'EMPLOY(DA(2),"ICC","P2",SE(X,1,30),DA(1),DA)
1004.21,2	END DATE	0;3	DATE Input transform: S 'NOT="E" D "NOT S X=Y K:Y<1 X LAST EDITED: OCT 18, 1987
1004.21,3	GROSS AMOUNT RECEIVED 0;4		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:+X'-X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1004.21,4	PAID	0;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:+X'-X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 25, 1987
1004.21,5	BASIS	0;6	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 25, 1987
1004.21,6	SPECIFY OTHER	0;7	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30:(SL(X)<1) X LAST EDITED: OCT 18, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/NOHHS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,14.01	HO ADDRESS	10;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 17, 1987
1004.2,14.02	HO CITY	10;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.03	HO STATE	10;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,14.04	HO ZIP CODE	10;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,14.05	POINT OF CONTACT	10;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.055	POC TITLE	10;10	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.06	POC PHONE	10;7	FREE TEXT ANSWER MUST BE 10-13 CHARACTERS IN LENGTH Input transform: K:\$L(X)>13!(\$L(X)<10) X LAST EDITED: SEP 15, 1987
1004.2,14.07	POC EXTENSION	10;8	FREE TEXT ANSWER MUST BE 1-4 CHARACTERS IN LENGTH Input transform: K:\$L(X)>4!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,14.08	POC AUTOVEN	10;9	FREE TEXT ANSWER MUST BE 7-13 CHARACTERS IN LENGTH Input transform: K:\$L(X)>13!(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1004.2,14.5	EMPLOYEE COST CENTER	2;4	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH Input transform: K:\$L(X)>6!(\$L(X)<1) X
1004.2,15	OCCUPATION	3;1	FOUNDER TO OCCUPATION FILE (#1001) (Required) LAST EDITED: SEP 15, 1987
1004.2,15.5	OCCUPATION CODE	3;2	FREE TEXT ANSWER MUST BE 5-6 CHARACTERS IN LENGTH Input transform: K:\$L(X)>6!(\$L(X)<5) X LAST EDITED: AUG 28, 1987
1004.2,16	TENURE	3;3	SET ; 'TEMP' FOR TEMPORARY ; 'NON' FOR NONCOMPETING ; 'CARE' FOR CAREER ; 'CARE' FOR CAREER-CONDITIONAL
1004.2,17	PAY PLAN	3;4	FREE TEXT EXAMPLES: (GS, WG, WD, GN) Input transform: K:\$L(X)>2!(\$L(X)<2)!('XZ2A) X
1004.2,18	PAY GRADE	3;5	FREE TEXT FORMAT: (01, 12, 09, 13) Input transform: K:\$L(X)>3!(\$L(X)<2) X

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRIMS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,19	PAY STEP	3;6	FREE TEXT ENTER THE LEVEL 01-99. Input transform: K:SL(X)>2!(\$L(X)<1) X
1004.2,20	PAY BASIS	3;7	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 1, 1987
1004.2,20.2	PAY INFORMATION	5;0	1004.21 SET (Add New Entry without Asking) Multiple 'IDENTIFIED' BY DATE IMPLEMENTED
1004.21,.01	TYPE OF PAY RECEIVED 0;1		SET (Required) ; '1' FOR REGULAR/BASE PAY ; '2' FOR CONTINUATION OF PAY ; '3' FOR COMPENSATION PAY ; '4' FOR ANNUAL LEAVE PAY ; '5' FOR SICK LEAVE PAY ; '6' FOR OTHER LEAVE PAY ; '7' FOR MISCELLANEOUS PAY LAST EDITED: OCT 18, 1987 CROSS-REFERENCE: 1004.2"P1 1)= S EMPLOY(DA(2),"ICC","P1",SE(X,1,30),DA(1),DA)="" 2)= K EMPLOY(DA(2),"ICC","P1",SE(X,1,30),DA(1),DA)
1004.21,1	DATE IMPLEMENTED	0;2	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: OCT 18, 1987 CROSS-REFERENCE: 1004.2"P2 1)= S EMPLOY(DA(2),"ICC","P2",SE(X,1,30),DA(1),DA)="" 2)= K EMPLOY(DA(2),"ICC","P2",SE(X,1,30),DA(1),DA)
1004.21,2	END DATE	0;3	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: OCT 18, 1987
1004.21,3	GROSS AMOUNT RECEIVED 0;4		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$",-) K:+X'->GL(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1004.21,4	RATE	0;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$",-) K:+X'->GL(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 25, 1987
1004.21,5	BASIS	0;6	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 25, 1987
1004.21,6	SPECIFY OTHER	0;7	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(\$L(X)<1) X LAST EDITED: OCT 18, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHIMS SYSTEM STORED IN 'EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,21	PAY RATE	3;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999 Input transform: S:X("\$" X-SP(X,"\$"),2) K:+X'-X(X'?.N1"."2N)!(X>99999)!(X<0) X LAST EDITED: SEP 2, 1987
1004.2,21.1	DOI SUBSISTENCE	3;16	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("\$" X-SP(X,"\$"),2) K:+X'-X(X'?.N1"."2N)!(X>99999)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,21.25	DOI SUBSISTENCE BASIS	3;17	SET ; 'H' FOR HOURLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,21.3	DOI QUARTERS	3;18	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("\$" X-SP(X,"\$"),2) K:+X'-X(X'?.N1"."2N)!(X>99999)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,21.35	DOI QUARTERS BASIS	3;19	SET; 'H' FOR HOURLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,21.4	DOI OTHER RATE	3;20	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("\$" X-SP(X,"\$"),2) K:+X'-X(X'?.N1"."2N)!(X>99999)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,21.45	DOI OTHER RATE BASIS	3;21	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,21.5	PAY STATUS	3;10	FOUNDER TO PAY STATUS FILE (#1093)
1004.2,21.6	DOI SPECIFY OTHER	3;22	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1004.2,22	SHIFT	3;9	SET ; '1' FOR DAY ; '2' FOR SWING ; '3' FOR GRAVEYARD
1004.2,23	MONTHS ON JOB	3;11	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 9999999999 Input transform: K:+X'-X!(X>999999999)!(X<0)!(X?.E1"."1N.N) X
1004.2,24	FEGLI CODE	3;12	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(SL(X)<1) X
1004.2,25	FEHB CODE	3;13	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(SL(X)<1) X
1004.2,26	PRIMARY SUPERVISOR	3;14	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRIMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,27	SECONDARY SUPERVISOR	3;15	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X
1004.2,30	PREMIUM PAY	12;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1004.2,31	PREMIUM PAY BASIS	12;2	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1004.2,32	NIGHT PAY	12;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1004.2,33	NIGHT PAY BASIS	12;4	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1004.2,34	SUNDAY PAY	12;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1004.2,35	SUNDAY PAY BASIS	12;6	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1004.2,36	OTHER PAY	12;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1004.2,37	OTHER PAY BASIS	12;8	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1004.2,38	OTHER PAY SPECIFY	12;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: OCT 17, 1987
1004.2,50	WITNESS INFORMATION	4;0	1004.22(Add New Entry without Asking) Multiple
1004.22,.01	WITNESS NAME	0;1	FREE TEXT ANSWER MUST BE 3-25 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>25!(SL(X)<3) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1004.22^B 1)= S EMPLOY(DA(2),"ICC",DA(1),4,"B",SE(X,1,30),DA)="" 2)= R EMPLOY(DA(2),"ICC",DA(1),4,"B",SE(X,1,30),DA)

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHMS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.22,1	STREET ADDRESS	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.22,2	CITY	0;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.22,3	STATE	0;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.22,4	ZIP CODE	0;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1004.22,5	DATE SIGNED	0;6	DATE Input transform: S %DT="E" D %DT S X-Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.22,6	WITNESS STATEMENT	1;0	1004.23 WORD-PROCESSING
1004.2,52	NARRATIVE	6;0	1004.24 WORD-PROCESSING
1004.2,53	CORRECTIVE ACTION	7;0	1004.25 WORD-PROCESSING
1004.2,54	REMARKS	8;0	1004.26 WORD-PROCESSING
1004.2,54.5	SAFETY REMARKS	11;0	1004.27 WORD-PROCESSING
1004.2,100	BODY PART CODE	100;1	POINTER TO BODY PART FILE (#1018) (Required)
1004.2,101	NATURE OF INJURY CODE	100;2	POINTER TO NOI FILE (#1017) LAST EDITED: SEP 15, 1987
1004.2,101.1	NATURE OF INJURY EXPLANATION	101;0	1004.28 WORD-PROCESSING
1004.2,101.2	OWCP USE-NOI CODE	100;18	FREE TEXT ANSWER MUST BE 1-5 CHARACTERS IN LENGTH Input transform: K:\$L(X)>5!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1004.2,102	INJURY OSHA CODE	100;3	POINTER TO OSHA INJURY CODES FILE (#1057)
1004.2,103	DATE OF DEATH	100;4	DATE XXXX-CAN'T BE ALTERED EXCEPT BY PROGRAMMER Input transform: S %DT="E" D %DT S X-Y K:Y<1 X I \$D(X) K:X<SP("ICC(DA,0),U,2) X LAST EDITED: SEP 14, 1987
1004.2,104	LOST TIME ACCIDENT	100;5	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,105	LOST WORK DAYS	100;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 999 Input transform: K:+X'-X1(X>999)!((X<0)!((X>E1)". "IN.N) X

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRIMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,106	LIGHT DUTY DAYS	100;7	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 999 Input transform: K:+X'-X1(X>999)!(X<0)!(X?.E1)".*1N.N) X
1004.2,107	NONFATAL LOST WORKDAYS	100;8	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,108	FIRST AID CASE	100;9	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,109	RECORDABLE CASE	100;10	SET ; 'Y' FOR YES ; 'N' FOR NO ; 'R' FOR REMOVED
1004.2,110	REPORTABLE CASE	100;11	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,111	TERMINATION TRANSFER	100;12	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,112	TYPE OF ACCIDENT	100;13	POINTER TO ACCIDENT FILE (#1025) LAST EDITED: SEP 15, 1987
1004.2,112.5	TYPE CODE	100;17	FREE TEXT ANSWER MUST BE 1-5 CHARACTERS IN LENGTH Input transform: K:SL(X)>5:(SL(X)<1) X LAST EDITED: AUG 28, 1987
1004.2,113	BUDG/SHIP	100;14	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15:(SL(X)<1) X
1004.2,114	SPECIFIC LOCATION	100;15	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:SL(X)>40:(SL(X)<1) X
1004.2,115	EVOLUTION OF INJURY	100;16	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20:(SL(X)<1) X
1004.2,116	EXTENT OF INJURY	110;1	SET;'1' FOR EQUALITY ; '2' FOR PERM TOTAL DISABILITY; '3' FOR PERM PARTIAL DISABILITY ; '4' FOR NO DISABILITY LIKELY ; '5' FOR MISSING LAST EDITED: DEC 23, 1987
1004.2,117	WORK OPERATION	110;2	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:SL(X)>40:(SL(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,118	SHIPS STATUS AT INJURY	110;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20:(SL(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,119	DAYS HOSPITALIZED	110;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 999 Input transform: K:+X'-X1(X>999)!(X<0)!(X?.E1)".*1N.N) X LAST EDITED: SEP 8, 1987
1004.2,120	CAUSE OF INJURY	110;5	POINTER TO CAUSE OF INJURY FILE (#1026) LAST EDITED: AUG 14, 1987
1004.2,120.1	CAUSE OF INJURY EXPLANATION	111;0	1004.29 WORD-PROCESSING

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSRRS/ACHIMS SYSTEM STORED IN 'EMPLOY'

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,121	WHO CAUSE OF ACCIDENT	110;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,122	FAILURE TO DO	110;7	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,123	WHY FAILURE	110;8	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,124	DUTY STATUS	120;1	SET (Required); 'ON' FOR ON DUTY ; 'OFF' FOR OFF DUTY LAST EDITED: AUG 14, 1987
1004.2,125	DATE SPEEDLETTER SENT	120;2	DATE Input transform: S ^DT="E" D ^DT S X-Y K:Y<1 X LAST EDITED: AUG 14, 1987
1004.2,126	SOURCE OF INJURY	120;3	POINTER TO SOURCE OF INJURY FILE (#1027) LAST EDITED: SEP 15, 1987
1004.2,126.5	SOURCE CODE	120;9	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH Input transform: K:SL(X)>6!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1004.2,127	AGENT OF ACCIDENT	120;4	POINTER TO AGENT OF ACCIDENT FILE (#1068) LAST EDITED: AUG 14, 1987
1004.2,128	AGT OF ACC PARTICULAR	120;5	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,129	UNSAFE PRACTICE	120;6	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:SL(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,130	UNSAFE CONDITION	120;7	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:SL(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,131	PPE INVOLVED INJURY	120;8	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:SL(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,132	STRESSOR INVOLVED INJURY	130;1	POINTER TO STRESSOR FILE (#1083) LAST EDITED: AUG 14, 1987
1004.2,133	TDY INJURY	130;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1004.2,134	RECREATION INJURY	130;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1004.2,135	VEHICLE INJURY	130;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHHS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,136	TWO WHEEL VEHICLE INJURY	130;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1004.2,137	OWCP FILE NUMBER	130;6	FREE TEXT ANSWER MUST BE 1-13 CHARACTERS IN LENGTH Input transform: K:SL(X)>13!(SL(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,138	JOB TRANSFER MED REASONS	130;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1004.2,139	RETIRED ON DISABILITY	130;8	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,140	THIRD PARTY CASE	130;9	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,141	CASE OUTCOME	130;10	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,141.1	THIRD PARTY	140;0	1004.3 Multiple
1004.3,.01	NAME OF THIRD PARTY	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1004.3"B 1)-6 EMPLOY(DM2),"IOC",DA(1),140,"B",SE(X,1,30),DA)="" 2)- K EMPLOY(DM2),"IOC",DA(1),140,"B",SE(X,1,30),DA)
1004.3,1	ADDRESS	0;2	FREE TEXT ANSWER MUST BE 1-70 CHARACTERS IN LENGTH Input transform: K:SL(X)>70!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.3,2	CITY	0;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.3,3	STATE	0;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.3,4	ZIP CODE	0;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.3,5	SETTLEMENT AMOUNT	0;6	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X" "\$ X-SP(X,"\$",2) K:+X'-W(X?7.NI"."2N)!(X>9999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,142	DATE TRANSFER FEHB TO OWCP	130;11	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: AUG 14, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHINS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,143	CLAIMS CLERK ID	130;12	FREE TEXT ANSWER MUST BE 1-3 CHARACTERS IN LENGTH Input transform: K:\$L(X)>3!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,144	INVESTIGATED BY	130;13	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1004.2,150	TICKLER FILE	150;0	1004.708 Multiple
1004.708,.01	TICKLER FILE	0;1	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 999999 (Multiply asked) Input transform: K:+X'-X!(X>999999)!(X<1)!(X?.E1)."IN.N) X LAST EDITED: SEP 25, 1987
1004.708,1	DATE TO DO	0;2	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: SEP 25, 1987
1004.708,2	RESPONSIBLE TO DO	0;3	FREE TEXT ENTER PERSON'S INITIALS Input transform: K:\$L(X)>3!(\$L(X)<1) X LAST EDITED: SEP 25, 1987
1004.708,3	ACTION TO TAKE	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: SEP 25, 1987
1004.708,4	REVERSE DATE	;	COMPUTED DATE MUMPS CODE: S Y(1004.708,4,1)= \$(SD(EMPLOY(DO,"IDC",DL,150,D2,0)):"(0),1:"") S X=99999999-\$P(Y(1004.708,4,1),U,2) ALGORITHM: 9999999-DATE TO DO
1004.2,200	WHERE FILED	200;1	SET (Required); 'OFF' FOR OFFICIAL PERSONNEL FOLDER; 'OACP' FOR OFFICE OF WORKER COMPENSATION PROGRAM LAST EDITED: AUG 14, 1987
1004.2,201	CLAIM RECORD CREATION DATE	200;2	DATE The date must be after the Date of Injury! Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: AUG 14, 1987
1004.2,202	DATE OF CLAIM	200;3	DATE XXX-CAN'T BE ALTERED EXCEPT BY PROGRAMMER Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: SEP 28, 1987
1004.2,202.5	CLAIM TYPE	200;12	POINTER TO CLAIM TYPE FILE (#1029) LAST EDITED: DEC 23, 1987
1004.2,203	SENT TO OACP	200;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: DEC 23, 1987
1004.2,204	DATE CLAIM SENT TO OACP	200;5	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: AUG 14, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSIRIS/ACHIMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,205	CLAIM CONROVERTED	200;6	SET (Required) ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 26, 1987
1004.2,205.1	CONROVERSION EXPLANATION	305;0	1004.31 WORD-PROCESSING
1004.2,205.2	CONROV SUSTAINED BY GWCP	200;11	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: OCT 31, 1987
1004.2,206	ADJUDICATION STATUS	13;0	1004.709POINTER Multiple
1004.709,.01	ADJUDICATION STATUS	0;1	POINTER TO ADJUDICATION STATUS FILE (#1061) (Required) (Multiply asked) LAST EDITED: JAN 12, 1988 CROSS-REFERENCE: 1004.709'B 1)= S EMPLOY(DA(2),"ICC",DA(1),13,"B",SE(X,1,30),DA)="" 2)= K EMPLOY(DA(2),"ICC",DA(1),13,"B",SE(X,1,30),DA)
1004.709,1	DATE ENTERED	0;2	DATE (Required) Input transform: S NOT="E" D NOT S X=Y K:Y<1 X
1004.709,2	DATE REMOVED	0;3	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: JAN 12, 1988
1004.2,207	FINAL DISPOSITION	200;8	SET ; 'A' FOR ACTIVE ; 'I' FOR INACTIVE LAST EDITED: AUG 14, 1987
1004.2,208	CLAIM FORM LAST RECEIVED	200;9	POINTER TO FORMS FILE (#1002) Input transform: S DIC("S")="I SP(^ (0),U,4)=1004" D DIC K DIC S DIO=OIE,X=Y K:Y<0 X CLAIM FORMS LAST EDITED: SEP 23, 1987
1004.2,209	CLAIM FORM NEXT	200;10	POINTER TO FORMS FILE (#1002) Input transform: S DIC("S")="I SP(^ (0),U,4)=1004" D DIC K DIC S DIO=OIE,X=Y K:Y<0 X SELECT CLAIM FORM LAST EDITED: SEP 24, 1987
1004.2,210	COP BEGIN DATE	220;1	DATE XL--CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! (Required) Input transform: S NOT="ET" D NOT S X=Y K:Y<1 X LAST EDITED: SEP 2, 1987
1004.2,210.1	COP NOT TO EXCEED DATE	220;2	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: OCT 31, 1987
1004.2,210.2	COP HOURS OF DAY	220;3	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'-X!(X>24)!(X<0)!(X?.E1". "IN.N) X LAST EDITED: OCT 31, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/NCIMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,210.3	COP RETURN TO WORK DATE	220;4	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: OCT 31, 1987
1004.2,210.4	COP RETURN TO WORK HOURS	220;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'=%I(X>24)!(X<0)!(X?.E1)".1N.N) X LAST EDITED: OCT 31, 1987
1004.2,210.5	COP RECURRENCE FLAG	220;6	SET;'Y' FOR YES ; 'N' FOR NO LAST EDITED: OCT 31, 1987
1004.2,210.6	COP DAYS USED	220;7	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 9999 Input transform: K:+X'=%I(X>9999)!(X<0)!(X?.E1)".1N.N) X LAST EDITED: OCT 31, 1987
1004.2,210.7	COP JOB ORDER COST	220;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$" %SP(X,"\$",2) K:+X'=%I(X'?.N1)".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 31, 1987
1004.2,229	D/T STOPPED WORK	221;1	DATE X%-CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: FEB 8, 1988
1004.2,229.1	WORK STOPPED BASE RATE	222;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$" %SP(X,"\$",2) K:+X'=%I(X'?.N1)".2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,229.2	WORK STOPPED BASE BASIS	222;2	SET;'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY;'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,229.3	WORK STOPPED SUBSISTENC	222;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$" %SP(X,"\$",2) K:+X'=%I(X'?.N1)".2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,229.35	WORK STOPPED SUB BASIS	222;4	SET;'H' FOR HOURLY;'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY;'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,229.4	WORK STOPPED QUARTERS	222;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$" %SP(X,"\$",2) K:+X'=%I(X'?.N1)".2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRHS SYSTEM STORED IN 'EMPLOY('

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,229.45	WORK STOPPED QUARTERS	222;6	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,229.5	WORK STOPPED OTHER RATE	222;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'-X(X'?..N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,229.55	WORK STOPPED OTHER BASIS	222;8	SET;'H' FOR HOURLY; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY;'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,229.6	WORK STOPPED OTHER SPECIFY	222;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1004.2,230	D/T PAY STOPPED	221;2	DATE XX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="ET" D ^%DT S %Y K:Y<1 X LAST EDITED: NOV 2, 1987
1004.2,231	D/T RETURNED TO WORK	221;3	DATE XX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="ET" D ^%DT S %Y K:Y<1 X LAST EDITED: NOV 2, 1987
1004.2,231.1	RETURNED TO WORK BASE RATE	231;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'-X(X'?..N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,231.2	RETURNED TO WORK BASE BASIS	231;2	SET;'H' FOR HOURLY; W FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY;'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,231.3	RETURN TO WORK SUBSISTENCE	231;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'-X(X'?..N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,231.35	RETURNED TO WORK SUB BASIS	231;4	SET;'H' FOR HOURLY; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: OCT 18, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE UCI: DIAL:[NCHIMS] OSHRS/NCHIMS SYSTEM
 STORED IN EMPLOY (31 ENTRIES)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,231.4	RETURN TO WORK QUARTERS	231;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] *SP(X,"\$",2) K:+X'-XL(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,231.45	RETURNED TO WORK QUR BASIS	231;6	SET;'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY ;'D' FOR DAILY LAST EDITED: OCT 18, 1987
1004.2,231.5	RETURN TO WORK OTHER RATE	231;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] *SP(X,"\$",2) K:+X'-XL(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,231.55	RETURNED TO WORK OTHER BASIS	231;8	SET;'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY ;'D' FOR DAILY LAST EDITED: OCT 18, 1987
1004.2,231.6	RETURN TO WORK SPECIFY OTHER	231;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1004.2,232	RATE RATE CHANGED	232;1	DWT Input transform: S *DT="E" D *DT S *Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,232.1	CHANGED BASE RATE	232;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] *SP(X,"\$",2) K:+X'-XL(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,232.2	CHANGED SUBSISTENCE	232;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] *SP(X,"\$",2) K:+X'-XL(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,232.3	CHANGED QUARTERS	232;4	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] *SP(X,"\$",2) K:+X'-XL(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,232.4	CHANGED OTHER RATE	232;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] *SP(X,"\$",2) K:+X'-XL(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,232.5	CHANGED OTHER SPECIFY	232;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,250	COP RECVD FROM		DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,250.5	COP RECVD THRU	250;2	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,250.6	COP RECVD GROSS	250;16	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) X:X'-X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1004.2,250.7	COP BASE RATE	250;17	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:X'-X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1004.2,250.8	COP BASE BASIS	250;18	SET: 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY, 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: OCT 18, 1987
1004.2,251	OTHER LEAVE FROM	250;7	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,251.5	OTHER LEAVE THRU	250;8	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,251.6	OTHER LEAVE SPECIFY	250;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1004.2,252	SICK LEAVE FROM	250;3	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,252.5	SICK LEAVE THRU	250;4	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,253	ANNUAL LEAVE FROM	250;5	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,253.5	ANNUAL LEAVE THRU	250;6	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,254	COP CLAIMED FROM	250;10	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,254.5	COP CLAIMED THRU	250;11	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,255	COMP CLAIM FROM	250;12	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,255.5	COMP CLAIMED THRU	250;13	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OS/VS/NOHMS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,255.6	SICK/ANNUAL LEAVE CLAIMED FROM 250;14		DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,255.7	SICK/ANNUAL LEAVE CLAIMED THRU 250;15		DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,256	COMP RECVD FROM	251;1	DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1004.2,256.1	COMP RECVD THRU	251;2	DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1004.2,256.2	COMP RECVD GROSS	251;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1".2N)!(X0999999.99)!(X0) X LAST EDITED: OCT 17, 1987
1004.2,256.3	COMPENSATION TYPE	251;4	SET; 'C' FOR COMPENSATION; 'S' FOR SCHEDULED AWARD LAST EDITED: NOV 2, 1987
1004.2,256.4	COMP BEGIN HOURS	251;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'=X1(X024)!(X0)!(X2.E1".1N.N) X LAST EDITED: NOV 2, 1987
1004.2,256.5	COMP END HOURS	251;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'=X1(X024)!(X0)!(X2.E1".1N.N) X LAST EDITED: NOV 2, 1987
1004.2,256.6	COMP LEAVE STATUS	251;7	SET; 'S' FOR SICK LEAVE; 'P' FOR ADMINISTRATIVE ; 'L' FOR LACOP ; 'T' FOR TEMP OFF SITION ; 'A' FOR ANNUAL LEAVE (VACATION) LAST EDITED: NOV 2, 1987
1004.2,256.7	COMP RECVD DATE	251;8	DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: NOV 2, 1987
1004.2,256.8	COMP RECURRENCE FLAG	251;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: NOV 2, 1987
1004.2,257	MISC PAY RECVD FROM	252;1	DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1004.2,257.1	MISC PAY RECVD THRU	252;2	DATE Input transform: S *DT="E" D *DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1004.2,257.2	MISC PAY RECVD GROSS	252;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1".2N)!(X0999999.99)!(X0) X LAST EDITED: OCT 17, 1987
1004.2,257.3	MISC RATE	252;4	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1".2N)!(X0999999.99)!(X0) X LAST EDITED: OCT 18, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHINS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,257.4	MISC RATE BASIS	252;5	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-WEEKLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: OCT 13, 1987
1004.2,257.5	MISC SPECIFY	252;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: OCT 18, 1987
1004.2,300	DEPENDENTS	300;1	SET; 'S' FOR SPOUSE; 'C' FOR CHILDREN; 'O' FOR OTHER; 'SC' FOR SPOUSE AND CHILDREN; 'SO' FOR SPOUSE & OTHER; 'SOC' FOR SPOUSE, CHILDREN, & OTHER ; 'CO' FOR CHILDREN AND OTHER ; 'N' FOR NONE LAST EDITED: NOV 12, 1987
1004.2,301	SICK/ANNUAL LEAVE CLAIMED	300;2	SET (Required) ; 'CO' FOR CONTINUATION OF REGULAR PAY & COMPENSATION IF DISABILITY GOES BEYOND 45 DAYS ; 'S/A' FOR SICK AND/OR ANNUAL LEAVE LAST EDITED: NOV 13, 1987
1004.2,302	REGULAR WORK HOURS	300;3	FREE TEXT ANSWER IN MILITARY TIME. EX: (0700-1500) Input transform: K:\$L(X)>14!(\$L(X)<6) X LAST EDITED: AUG 20, 1987
1004.2,303	REGULAR WORK SCHEDULE	300;4	FREE TEXT EXAMPLE: (M-F, or M,TU,W,TH,F) Input transform: K:\$L(X)>15!(\$L(X)<3) X LAST EDITED: AUG 20, 1987
1004.2,303.5	REGULAR NUMBER OF HOURS DAILY	300;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 24 Input transform: K:+X'=X! (X>24)! (X<1)! (X?.EL)". "IN.N) X LAST EDITED: OCT 31, 1987
1004.2,303.5	PARTTIME WORK HOURS PER DAY	300;7	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 8 Input transform: K:+X'=X! (X>8)! (X<1)! (X?.EL)". "IN.N) X LAST EDITED: OCT 31, 1987
1004.2,304	DATE RECEIVED NOTICE	300;5	DATE XX--CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S \$UT="E" D "NOT S X=Y R:Y<1 X LAST EDITED: NOV 2, 1987
1004.2,307	INJURED IN LINE OF DUTY	300;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1004.2,307.1	EXPLAIN INJURY	306;0	1004.32 WORD-PROCESSING
1004.2,308	WILLFUL MISCONDUCT	300;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1004.2,308.1	EXPLAIN WILLFUL MISCONDUCT	301;0	1004.33 WORD-PROCESSING

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHHS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,310	FIRST EXAM DATE	303;1	DATE XX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: NOV 2, 1987
1004.2,311	DISABLED FOR WORK	303;2	SET ; 'Y' FOR YES ; 'N' FOR NO
1004.2,312	SUPS FACIS AGREE WITH EMP	303;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1004.2,312.1	SUPERVISOR FACIS	304;0	1004.34 WORD-PROCESSING
1004.2,314	CLAIM FILING CATEGORY	303;4	SET; '1' FOR NO LOST TIME AND NO MEDICAL EXPENSE; '2' FOR NO LOST TIME, MEDICAL EXPENSE INCURRED; '3' FOR LOST COVERED BY LEAVE, INOP, OR CCP LAST EDITED: AUG 20, 1987
1004.2,315	RELATES TO ILLNESS	307;0	1004.35 WORD-PROCESSING
1004.2,316	WHY FILED AFTER 30 DAYS	308;0	1004.36 WORD-PROCESSING
1004.2,317	REASON FOR STATEMENT DELAY	309;0	1004.37 WORD-PROCESSING
1004.2,318	WHY MEDICAL REPORTS DELAYED	310;0	1004.38 WORD-PROCESSING
1004.2,319	DATE LAST EXPOSED TO CONDITION	303;5	DATE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1004.2,320	DESCRIBE NEW WORK ASSIGNMENT	311;0	1004.39 WORD-PROCESSING
1004.2,321	DATE 1ST REALIZED	303;6	DATE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1004.2,322	DATE EMP/CLAIMANT SIGNED	303;7	DATE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1004.2,324	NEW WORK WEEK	303;8	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<3) X LAST EDITED: AUG 20, 1987
1004.2,329	DUTY CHANGED DUE TO DISABILITY	303;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1004.2,329.1	DESCRIBE CHANGED DUTY	314;0	1004.4 WORD-PROCESSING
1004.2,330	DATE HEALTH DEDUCTION RESUMED	303;10	Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHHS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,331	DATE OPT INS DEDUCTION RESUMED	303;11	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1004.2,332	DATE HEALTH DEDUCT STOPPED	303;12	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 4, 1987
1004.2,333	NEW HEALTH PLAN DATE	303;13	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1004.2,400	ATTENDING PHYSICIAN	400;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,400.1	DOCS ADDRESS	400;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 17, 1987
1004.2,400.2	DOCS CITY	400;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,400.3	DOCS STATE	400;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,400.4	DOCS ZIPCODE	400;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,400.5	DOCS PHONE NUMBER	400;6	FREE TEXT ANSWER MUST BE 7-14 CHARACTERS IN LENGTH Input transform: K:\$L(X)>14!(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1004.2,400.6	DOES DOCTOR SPECIALIZE	400;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 15, 1987
1004.2,400.7	SPECIALITY	400;8	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,400.8	TAX ID	400;9	FREE TEXT ANSWER MUST BE 1-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,400.9	DATE SIGNED	400;10	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.2,401	REFERRAL PHYSICIAN	401;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,401.1	RP ADDRESS	401;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 17, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OGRS/ACHMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,401.2	RP CITY	401;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,401.3	RP STATE	401;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,401.4	RP ZIPCODE	401;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,401.5	RP PHONE NUMBER	401;6	FREE TEXT ANSWER MUST BE 7-14 CHARACTERS IN LENGTH Input transform: K:SL(X)>141(SL(X)<7) X LAST EDITED: SEP 15, 1987
1004.2,401.6	DOES REF DOCTOR SPECIALIZE	401;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 15, 1987
1004.2,401.7	RP SPECIALTY	401;8	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,401.8	RP TAX ID	401;9	FREE TEXT ANSWER MUST BE 1-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,401.9	RP DATE SIGNED	401;10	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.2,402	SUPERVISOR'S NAME	402;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,402.1	SUPERVISOR'S TITLE	402;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,402.2	SUPERVISOR'S PHONE NUMBER	402;3	FREE TEXT ANSWER MUST BE 7-14 CHARACTERS IN LENGTH Input transform: K:SL(X)>141(SL(X)<7) X LAST EDITED: SEP 15, 1987
1004.2,402.3	SUPERVISOR'S AUTO/VN	402;4	FREE TEXT ANSWER MUST BE 7-8 CHARACTERS IN LENGTH Input transform: K:SL(X)>81(SL(X)<7) X LAST EDITED: SEP 15, 1987
1004.2,402.4	DATE SUPERVISOR SIGNED	402;5	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.2,402.6	SUPERVISORY ACC REPORT	402;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: DEC 9, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,502	SPOUSE AT DEATH	501;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,502.1	ADDRESS OF SPOUSE	501;2	FREE TEXT ANSWER MUST BE 1-55 CHARACTERS IN LENGTH Input transform: K:SL(X)>55!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,502.2	CITY OF SPOUSE	501;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,502.3	STATE OF SPOUSE	501;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,502.4	ZIPCODE OF SPOUSE	501;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,503	OTHER DEPEND AT DEATH	502;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: AUG 21, 1987
1004.2,504	OTHER DEPEND ADDRESS AT DEATH	502;1	FREE TEXT ANSWER MUST BE 1-55 CHARACTERS IN LENGTH Input transform: K:SL(X)>55!(SL(X)<1) X LAST EDITED: SEP 9, 1987
1004.2,504.1	OTHER DEPEND CITY	502;2	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:SL(X)>23!(SL(X)<1) X LAST EDITED: SEP 9, 1987
1004.2,504.2	OTHER DEPEND STATE	502;3	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 11, 1987
1004.2,504.3	OTHER DEPEND ZIP CODE	502;4	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 9, 1987
1004.2,505	SERVICE NO	500;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: AUG 21, 1987
1004.2,506	BRANCH OF SERVICE	500;4	FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: AUG 21, 1987
1004.2,507	PERIOD OF SERVICE	500;5	FREE TEXT FORMAT: MO/YR TO MO/YR Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 8, 1987
1004.2,515	TOTAL BURIAL EXPENSE	505;1	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 9999999 Input transform: K:+X'-X1(X9999999)!(X<0)!(X>.E1)".IN.N) X LAST EDITED: AUG 21, 1987

DDIA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHMS SYSTEM STORED IN EMPLOY(

DDIA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,516	BURIAL EXPENSE PAID BY VA 505;2		NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 99999999 Input transform: K:+X'~K1(X>99999999)!(X<0)!(X2.E1". "IN.N) X LAST EDITED: AUG 21, 1987
1004.2,517	PARTY PAYING FOR BURIAL 506;1		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,517.01	ADDRESS OF PERSON PAYING 506;2		FREE TEXT ANSWER MUST BE 1-45 CHARACTERS IN LENGTH Input transform: K:\$L(X)>45!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,517.02	CITY OF PER PAYING 506;3		FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,517.03	STATE OF PERSON PAYING 506;4		FOINDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,517.04	ZIP CODE OF PER PAYING 506;5		FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,517.1	AMOUNT OTHER PARTY PAID BURIAL 505;3		NUMBER TYPE A NUMBER BETWEEN .01 AND 99999999.99 Input transform: S:X["\$"] X~SP(X,"\$"),2) K:+X'~K1(X'7.N1". "2N)!(X>99999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987
1004.2,520	DOCS NARRATIVE OF REL DISEASE 507;0		1004.41 WORD-PROCESSING
1004.2,521	DOCS TREATMENT/DIAG 508;0		1004.42 WORD-PROCESSING
1004.2,522	TREATMENT PRIOR TO DEATH 509;0		1004.43 WORD-PROCESSING
1004.2,523	TREATMENT DATES (PRE-DEATH) 505;6		DATE Input transform: S ~DT="E" D ~DT S X~Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,524	DIRECT CAUSE OF DEATH 510;0		1004.44 WORD-PROCESSING
1004.2,525	CONTRIBUTORY CAUSE OF DEATH 511;0		1004.45 WORD-PROCESSING
1004.2,526	WAS DEATH DUE TO WORK 505;7		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OCHRS/ACHRS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,526.1	EXPLAIN CAUSAL RELATIONSHIP	512;0	1004.46 WORD-PROCESSING
1004.2,527	AUTOPSY/BIOPSY PERFORMED	505;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,530	SPOUSE DOB	501;6	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,531	DATE OF MARRIAGE	515;1	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,532	LIVING W/SPOUSE AT DEATH	515;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,533	SPOUSE PREVIOUSLY MARRIED	515;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHMS SYSTE! STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,536	CHILDREN PRIOR MARRIAGE	517;0	1004.48 Multiple
1004.48,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1004.48*B 1)= S EMPLOY(DA(2),"IOC",DA(1),517,"B",SE(X,1,30),DA)="" 2)= K EMPLOY(DA(2),"IOC",DA(1),517,"B",SE(X,1,30),DA)
1004.48,1	RELATIONSHIP	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.48,2	DATE OF BIRTH	0;3	DATE Input transform: S WDT="E" D WDT S X-Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.48,3	ADDRESS	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.48,4	CITY	0;5	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.48,5	STATE	0;6	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.48,6	ZIP CODE	0;7	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,537	CHILD & LEGAL GUARDIAN	518;0	1004.49 Multiple
1004.49,.01	CHILD'S NAME	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1004.49*B 1)= S EMPLOY(DA(2),"IOC",DA(1),518,"B",SE(X,1,30),DA)="" 2)= K EMPLOY(DA(2),"IOC",DA(1),518,"B",SE(X,1,30),DA)
1004.49,1	GUARDIAN NAME	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.49,2	GUARDIAN'S ADDRESS	0;3	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X LAST EDITED: SEP 15, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRIMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.49,3	GUARDIAN'S CITY	0;4	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.49,4	GUARDIAN'S STATE	0;5	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.49,5	GUARDIAN'S ZIPCODE	0;6	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1988
1004.2,538	OTHER DEPENDENTS	519;0	1004.5 Multiple
1004.5,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1004.5"B 1)= S EMPLOY(DA(2),"ICC",DA(1),519,"B",SE(X,1,30),DA)="" 2)= K EMPLOY(DA(2),"ICC",DA(1),519,"B",SE(X,1,30),DA)
1004.5,1	RELATIONSHIP	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.5,2	DATE OF BIRTH	0;3	DATE Input transform: S NOT="E" D NOT S X Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.5,3	ADDRESS	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.5,4	CITY	0;5	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.5,5	STATE	0;6	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.5,6	ZIPCODE	0;7	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,539	OTHER DEPEND DOB	515;5	DATE Input transform: S NOT="E" D NOT S X Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,540	OTHER DEPEND OCCUPATION	515;6	POINTER TO OCCUPATION FILE (#1001) LAST EDITED: SEP 11, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHMS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,541	OTHER DEPEND SSN	515;7	FREE TEXT ANSWER MUST BE 9 CHARACTERS IN LENGTH Input transform: K:SL(X)>9!(SL(X)<9)!'(X?9N) X LAST EDITED: SEP 11, 1987
1004.2,542	OTHER DEPEND RELATIONSHIP	515;8	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<3) X LAST EDITED: AUG 21, 1987
1004.2,543	PARTIAL/TOTAL DEPENDENCY	515;9	SET ; 'T' FOR TOTAL ; 'P' FOR PARTIAL LAST EDITED: SEP 9, 1987
1004.2,544	AMT CONTRIB DEP PRIOR TO DEATH	515;10	NUMBER TYPE A NUMBER BETWEEN .01 AND 9999999.99 Input transform: S:X('\$' X=SP(X,'\$',2) K:+X'=X(X'7.NI'."2N)!(X9999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987
1004.2,545	DEP W/EMP 12 MOS PRIOR TO DIH	520;1	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,546	TOTAL ROOM/BOARD EXTRA	520;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999 Input transform: S:X('\$' X=SP(X,'\$',2) K:+X'=X(X'7.NI'."2N)!(X99999)!(X<0) X LAST EDITED: AUG 24, 1987
1004.2,546.1	ROOM/BOARD BASIS	520;5	SET; 'W' FOR WEEKLY; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 10, 1987
1004.2,547	VALUE RM/ED IF NO FIXED AMT	520;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X('\$' X=SP(X,'\$',2) K:+X'=X(X'7.NI'."2N)!(X9999999.99)!(X<0) X LAST EDITED: AUG 24, 1987
1004.2,547.1	VALUE BASIS	520;4	SET; 'W' FOR WEEKLY; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 10, 1987
1004.2,548	TYPE OF WORK PERFORMED	523;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,548.1	PERIOD OF EMPLOYMENT	523;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,548.2	MONTHLY PAY RATE	523;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X('\$' X=SP(X,'\$',2) K:+X'=X(X'7.NI'."2N)!(X9999999.99)!(X<0) X LAST EDITED: SEP 15, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHHS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,548.3	NAME OF EMPLOYER	523;4	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,548.4	ADDRESS OF EMPLOYER	523;5	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,548.5	CITY OF EMPLOYER	523;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,548.6	STATE OF EMPLOYER	523;7	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,548.7	ZIP CODE OF EMPLOYER	523;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,549	INVESTMENTS	524;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999999.99 Input transform: S:X("\$" X=SP(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>99999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,549.1	PENSIONS OF OTHER DEPENDENT	524;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999999.99 Input transform: S:X("\$" X=SP(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>99999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,549.2	PERSONS OTHER THAN EMPLOYEE	524;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999999.99 Input transform: S:X("\$" X=SP(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>99999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,549.3	OTHER INCOME FOR DEPENDENTS	524;4	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999999.99 Input transform: S:X("\$" X=SP(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>99999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,549.4	TOTAL DEPENDENTS OTHER INCOME	524;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999999.99 Input transform: S:X("\$" X=SP(X,"\$",2) K:+X'-X(X'?.N1"."2N)!(X>999999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,550	DEPEND SPOUSE DOR	520;6	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,551	DEPEND SPOUSE OCCUPATION	520;7	POINTER TO OCCUPATION FILE (#1001) LAST EDITED: SEP 11, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS, NCHMS SYSTEM SIGNED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,552	DEPEND SPOUSE MONTHLY WAGE 520;8		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1004.2,553	DEPEND INCOME 12 MOS PRIOR 520;9		NUMBER TYPE A NUMBER BETWEEN .01 AND 99999999.99 Input transform: S:X["\$"] X=SP(X,"\$",2) K:+X'=X(X'?..N1". "2N)!(X>99999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987
1004.2,554	DEPENDENTS PROPERTY 521;0		1004.51 Multiple
1004.51,.01	DEPENDENTS PROPERTY 0;1		FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:\$L(X)>40!(\$L(X)<1) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1004.51.B 1)= S EMPLOY(DA(2),"ICC",DA(1),521,"B",SE(X,1,30),DA)="" 2)= K EMPLOY(DA(2),"ICC",DA(1),521,"B",SE(X,1,30),DA)
1004.51,1	DATE ACQUIRED 0;2		DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: SEP 15, 1987
1004.51,2	VALUE 0;3		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999999.99 Input transform: S:X["\$"] X=SP(X,"\$",2) K:+X'=X(X'?..N1". "2N)!(X>999999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1004.2,555	ADDRESS OF PERSON SIGNING CASED	522;1	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LGTH
1004.2,555.1	CITY OF PERSON SIGNING CASED	522;2	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LGTH
1004.2,555.2	STATE OF PERSON SIGNING CASED	522;3	FOUNDER TO STATE FILE #(1003)
1004.2,555.3	ZIP CODE OF PERSON	522;4	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LGTH

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRIMS SYSTEM SACRED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,603	CASE OF DEATH	600;4	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS
1004.2,605	EMPLOYED IN DUTY 11 MOS 600;2		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,606	DUTY AFFORDED 11 MOS WORK 600;3		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,609	ATTORNEY NAME	601;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,609.1	ATTORNEY ADDRESS	601;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>541(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,609.2	ATTORNEY CITY	601;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,609.3	ATTORNEY STATE	601;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,609.4	ATTORNEY ZIP CODE	601;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,610	CLAIM FILED WORK	600;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,611	NEXT OF KIN	602;1	FREE TEXT ANSWER MUST BE 1-35 CHARACTERS IN LENGTH Input transform: K:SL(X)>351(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,611.1	ADDRESS OF NEXT OF KIN 602;2		FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>541(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,611.2	CITY OF NEXT OF KIN 602;3		FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,611.3	STATE OF NEXT OF KIN 602;4		FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1004.2,611.4	ZIP CODE OF NEXT OF KIN 602;5		FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 15, 1987
1004.2,611.5	RELATIONSHIP OF NEXT OF KIN 602;6		FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 15, 1987
1004.2,612	LOCATION OF DEATH	600;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 4, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHHS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,701	DRIE WAGES LOST FROM	700;7	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 9, 1987
1004.2,701.5	DRIES WAGES LOST THRU	700;9	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 9, 1987
1004.2,702	HRS COMP CLAIMED	700;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 999 Input transform: K:+X'~X!(X>999)!(X<0)!(X?.E1"."N.N) X LAST EDITED: AUG 25, 1987
1004.2,703	SCHEDULE AWARD	700;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,704	WAS ANY PAY RECEIVED	700;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,708	APPLY/RECVD VA BENEFITS	700;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,710	VA CLAIM NUMBER	701;1	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,710.1	VA OFFICE WHERE CLAIM FILED	701;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,710.2	VA STREET ADDRESS	701;3	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,710.3	VA CITY	701;4	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,710.4	VA STATE	701;5	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1004.2,710.5	VA ZIP CODE	701;6	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 16, 1987
1004.2,710.6	VA NATURE OF DISABILITY	701;7	FREE TEXT ANSWER MUST BE 1-57 CHARACTERS IN LENGTH Input transform: K:SL(X)>57!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,710.7	VA AMOUNT OF PAYMENT	701;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999.99 Input transform: S:X["\$" X=SP(X,"\$",2) K:+X'~X(X?7.N1"."2N)!(X>99999.99)!(X<0) X LAST EDITED: SEP 16, 1987
1004.2,713	ANY OTHER RETIRE/DISABILITY	700;10	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHINS SYSTEM STORED IN 'EMPLOY'

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,714	OTHER ANNUITIES CLAIM NUMBER	707;1	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,714.1	OTHER ANNUITY DATE BEGAN	707;2	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,714.2	OTHER ANNUITY AMOUNT	707;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:~X'~X(X'7.NI".2N)!(X>9999999.99)!(X<0) X LAST EDITED: SEP 16, 1987
1004.2,714.3	OTHER ANNUITY OFFICE	707;4	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,714.4	OTHER ANNUITY STREET ADDRESS	707;5	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,714.5	OTHER ANNUITY CITY	707;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,714.6	OTHER ANNUITY STATE	707;7	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1004.2,714.7	OTHER ANNUITY ZIP CODE	707;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(\$L(X)<5) X LAST EDITED: SEP 16, 1987
1004.2,720	OFFICIAL REMARKS	402;6	FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:SL(X)>50!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,722	DEPENDENT INFO	706;0	1004.7(Add New Entry without Asking) Multiple
1004.7,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.7,1	BIRTHDATE	0;2	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 25, 1987
1004.7,2	RELATIONSHIP	0;3	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(\$L(X)<1) X LAST EDITED: AUG 25, 1987
1004.7,3	LIVING WITH YOU	0;4	SET ; 'YES' FOR YES ; 'NO' FOR NO LAST EDITED: SEP 16, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHIMS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.7,5	SUPPORT PAYMENTS?	0;6	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 16, 1987
1004.7,6	ORDERED BY THE COURT?	0;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 16, 1987
1004.7,7	AMOUNT OF PAYMENTS	0;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999.99 Input transform: S:X("\$") X-SP(X,"\$",2) K:+X'-X(X'?N1"."2N)!(X>9999.99)!(X<0) X LAST EDITED: SEP 9, 1987
1004.7,8	PER	0;9	SET ; 'M' FOR MONTHLY ; 'W' FOR WEEKLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY LAST EDITED: SEP 9, 1987
1004.7,9	STREET ADDRESS	0;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 9, 1987
1004.7,10	CITY	0;10	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 9, 1987
1004.7,11	STATE	0;11	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 9, 1987
1004.7,12	ZIP CODE	0;12	FREE TEXT FORMAT: XXXXX-XXXX OR XXXXX Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 9, 1987
1004.2,727	TOTAL YRS CIV SERVICE	704;2	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 50 Input transform: K:+X'-X!(X>50)!(X<1)!(X?.E1"."1N.N) X LAST EDITED: SEP 9, 1987
1004.2,727.5	TOTAL MOS CIV SERVICE	700;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 11 Input transform: K:+X'-X!(X>11)!(X<1)!(X?.E1"."1N.N) X LAST EDITED: SEP 9, 1987
1004.2,728	EMP ENROLLED IN FEHB	704;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,728.1	PAY PERIOD END DATE FOR FEHB	700;6	DATE Input transform: S :DT-"E" D "DT S X-Y K:Y<1 X LAST EDITED: SEP 9, 1987
1004.2,729	EMP ENROLLED IN OPT LIFE INS	704;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,729.1	LIFE INS OPTION	704;5	SET ; 'A' FOR A ; 'B' FOR B ; 'C' FOR C LAST EDITED: SEP 28, 1987
1004.2,729.2	HOW MANY OPT/B MULTIPLES	704;6	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<3) X LAST EDITED: AUG 21, 1987

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRHS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,731	PERSON TO CONTACT	700;1	SET ; 'S' FOR SUPERVISOR ; 'O' FOR OTHER LAST EDITED: SEP 16, 1987
1004.2,801	REFURCHASE LEAVE	704;1	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,802	STATE REASON FOR NOT WORKING 801;0		1004.52 WORD-PROCESSING
1004.2,806	PAY REC'D DURING COMP	704;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,808	COMMISSION & SELF-EMP	812;0	1004.701(Add New Entry without Asking) Multiple
1004.2,701,.01	DRIES AND HOURS WORKED 0;1		FREE TEXT FORMAT: MO/YR-MO/YR XXX HRS (Multiply asked) Input transform: K:\$L(X)>201(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004..701,1	NAME OF BUSINESS	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1004.701,1.1	STREET ADDRESS	0;6	FREE TEXT ANSWER MUST BE 1-45 CHARACTERS IN LENGTH Input transform: K:\$L(X)>451(\$L(X)<1) X LAST EDITED: SEP 8, 1987
1004.701,1.2	CITY	0;7	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:\$L(X)>231(\$L(X)<1) X LAST EDITED: SEP 8, 1987
1004.701,1.3	STATE	0;8	FOUNDER TO STATE FILE (#1003) LAST EDITED: NOV 12, 1987
1004.701,1.4	ZIP CODE	0;9	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>101(\$L(X)<5) X LAST EDITED: SEP 8, 1987
1004.2,701,2	SELF-EMPLOYED OR COMMISSION 0;3		SET ; 'S' FOR SELF-EMPLOYED ; 'C' FOR COMMISSION LAST EDITED: SEP 8, 1987
1004.701,3	TYPE OF ACTIVITY PERFORMED 0;4		FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>201(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1004.701,4	INCOME DERIVED	0;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: AUG 23, 1987
1004.2,811	TRAINING REGISTRATION NUMBER 804;1		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: SEP 16, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACRHS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,811.1	TRAINING STREET ADDRESS	804;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,811.2	TRAINING CITY	804;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,811.3	TRAINING STATE	804;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1004.2,811.4	TRAINING ZIP CODE	804;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 16, 1987
1004.2,811.5	TRAINING DATE OF REGISTRATION	804;6	DATE Input transform: S %DT="E" D ^DT S %-Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,812	EMP RECEIVED MONIES FROM OACP	704;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1004.2,816	EXPLAIN CHS IN EMP HLTH/LIFE	805;0	1004.53 WORD-PROCESSING
1004.2,1601	TREATMENT/EXAM FOR CAUSE	1600;1	SET ; 'A' FOR AGREE TO FEE SCHEDULE ; 'B' FOR FURNISH NECESSARY TREATMENT NOT SURGERY; 'C' FOR EXAM FOR JOB RELATED CAUSE & TREATMENT LAST EDITED: AUG 21, 1987
1004.2,1604	MEDICAL CARE FACILITY	1601;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,1604.1	MED CARE STREET ADDRESS	1601;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,1604.2	MED CARE CITY	1601;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,1604.3	MED CARE STATE	1601;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1004.2,1604.4	MED CARE ZIP CODE	1601;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 16, 1987
1004.2,1605	HISTORY EMPLOYEE GAVE DOCTOR	1602;0	1004.54 WORD-PROCESSING

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHA/ACRHS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,1606	PRES/PRIOR CONDITION	1600;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1606.1	EXPLAIN INJ/DIS/IMP	11603;0	1004.55 WORD-PROCESSING
1004.2,1607	DOC BELIEVE INJ WORK-REL	1600;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1607.1	DOC EXPLAIN-NOT WORK RELATED	1604;0	1004.56 WORD-PROCESSING
1004.2,1608	REQUIRE HOSPITALIZATION	1600;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1608.1	ADMISSION DATE	1600;6	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1608.2	DISCHARGE DATE	1600;7	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1609	ADDITIONAL HOSPITALIZATION REQ	1600;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1610	SURGERY REQUIRED	1600;9	FREE TEXT ANSWER MUST BE 2-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<2) X LAST EDITED: AUG 21, 1987
1004.2,1611	SURGERY DATE	1600;10	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1612	OTHER TREATMENT PROVIDED	1605;0	1004.57 WORD-PROCESSING
1004.2,1613	PERMANENT EFFECTS	1606;0	1004.58 WORD-PROCESSING
1004.2,1614	DATE OF 1ST EXAM	1600;11	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1615	TREATMENT DATES	1607;0	1004.702 DATE (Add New Entry without Asking) Multiple
1004.702,.01	TREATMENT DATES	0;1	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,1616	DATE DISCHARGED FROM TREATMENT	1600;12	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1617	DATE RESUME LIGHT WORK	1600;13	DATE Input transform: S ^DT="E" D ^NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987

DATA DICTIONARY #1004.2 — INQUIRY COMP AND CLAIMS SUB-FILE OSHRS/NCIMS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,1618	DATE RESUME REG WORK	1600;14	DATE Input transform: S ^DT="E" D ^DT S ^-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1619	EXPLAIN EMP WORK LIMITATIONS	1608;0	1004.59 WORD-PROCESSING
1004.2,1623	AUTHORIZING OFFICIAL NAME	1610;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,1623.1	AUTHORIZING OFFICIAL TITLE	1610;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.2,1701	DATE OF EXAM	1700;14	DATE Input transform: S ^DT="E" D ^DT S ^-Y K:Y<1 X LAST EDITED: SEP 9, 1987
1004.2,1702	JOB REQUIRES DRIVING	1700;1	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1702.1	SPECIFY VEHICLE	1700;9	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: AUG 28, 1987
1004.2,1703	JOB REQ OPERATING MACHINERY	1700;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1703.1	SPECIFY MACHINERY	1700;10	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: AUG 28, 1987
1004.2,1704	HRS PER DAY EMP WORKS	1700;3	NUMBER TYPE A NUMBER BETWEEN 1 AND 8 Input transform: K:+X'=X!(X>8)!(X<1)!(X?.E1)". "N.N) X LAST EDITED: AUG 21, 1987
1004.2,1705	DAYS PER WEEK EMP WORKS	1700;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 5 Input transform: K:+X'=X!(X>5)!(X<1)!(X?.E1)". "N.N) X LAST EDITED: AUG 21, 1987
1004.2,1706	DOES DOC AGREE W/ EMP STATEMENT	1700;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1706.1	EXPLAIN WHAT EMP TOLD YOU	1701;0	1004.706 WORD-PROCESSING
1004.2,1707	DESCRIBE CLINICAL FINDINGS	1702;0	1004.6 WORD-PROCESSING
1004.2,1708	DIAGNOSIS (INQUIRY)	1703;0	1004.61 WORD-PROCESSING

DATA DICTIONARY #1004.2 -- INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHRS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,1709	DIAGNOSIS (OTHER COND)	1704;0	1004.62 WORD-PROCESSING
1004.2,1710	CAN EMP RETURN TO REG WORK	1700;6	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1710.1	FULL OR PART-TIME	1700;7	SET ; 'F' FOR FULL TIME ; 'P' FOR PART TIME ; 'f' FOR FULL TIME ; 'p' FOR PART TIME LAST EDITED: AUG 21, 1987
1004.2,1710.11	PART-TIME HRS PER DAY	1700;16	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 8 Input transform: K:+X'-X!(X>8)!(X<1)!(X?.E1)". "IN.N) X LAST EDITED: AUG 21, 1987
1004.2,1712	DESCRIBE FURTHER RESTRICTIONS	1706;0	1004.707 WORD-PROCESSING
1004.2,1713	TOTAL DISABILITY FROM	1707;1	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,1713.1	TOTAL DISABILITY IF	1707;2	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,1714	PARTIAL DISABILITY FROM	1707;3	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,1714.1	PARTIAL DISABILITY TO	1707;4	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 16, 1987
1004.2,1715	ADVISED EMP CAN RESUME WORK	1700;11	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,1715.1	DATE OF ADVICE	1700;12	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1716	NEXT EXAM DATE	1700;13	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1004.2,1720	USUAL WORK ACTIVITY	1709;0	1004.703 FORMER Multiple
1004.703,.01	USUAL WORK ACTIVITY	0;1	FOUNDER TO ACTIVITIES FILE (#1007) (Multiply asked) LAST EDITED: SEP 16, 1987
1004.703,1	DESCRIPTION OF ACTIVITY	0;2	FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:SL(X)>50!(SL(X)<1) X LAST EDITED: NOV 12, 1987
1004.703,2	CONT/INTER	0;3	SET ; 'C' FOR CONTINUOUS ; 'I' FOR INTERMITTENT LAST EDITED: SEP 9, 1987

DIPA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.703,3	HOURS PER DAY	0;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 10 Input transform: K:+X'-X1(X>10)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1004.703,4	DEGREES F	0;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 200 Input transform: K:+X'-X1(X>200)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1004.703,5	DEA	0;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 150 Input transform: K:+X'-X1(X>150)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1004.2,1721	ALTERED WORK ACTIVITY 1721;0		1004.704FOINIER Multiple
1004.704,.01	ALTERED WORK ACTIVITY 0;1		FOINIER TO ACTIVITIES FILE (#1007) (Multiply asked) LAST EDITED: SEP 16, 1987
1004.704,1	DESCRIBE ACTIVITY 0;2		FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:\$L(X)>50!(\$L(X)<1) X LAST EDITED: NOV 12, 1987
1004.704,2	CONT/INTER	0;3	SET ; 'C' FOR CONTINUOUS ; 'I' FOR INTERMITTENT LAST EDITED: SEP 9, 1987
1004.704,3	HOURS PER DAY	0;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 10 Input transform: K:+X'-X1(X>10)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1004.704,4	DEGREES F	0;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 200 Input transform: K:+X'-X1(X>200)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1004.704,5	DEA	0;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 150 Input transform: K:+X'-X1(X>150)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1004.2,2003	DATE OF MOST RECENT EXAM 2006;1		DATE Input transform: S ~DT="E" D ~DT S X-Y K:Y<1 X LAST EDITED: AUG 27, 1987
1004.2,2004	EMP TOTALLY DISABLED FOR WORK 2006;2		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,2005	DESCRIBE PRESENT IMPAIRMENT 2007;0		1004.63 WORD-PROCESSING

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHIMS SYSTEM STORED IN 'EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,2006	20A DIAGNOSIS	2008;0	1004.64 WORD-PROCESSING
1004.2,2007	TREATMENT/FREQUENCY OF ILL/INJ	2009;0	1004.65 WORD-PROCESSING
1004.2,2009	UNRELATED DISABILITIES	2010;0	1004.66 WORD-PROCESSING
1004.2,2010	WILL DISABIL GO EAST 90 DAYS	2006;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1004.2,2010.1	EXPECTED DATE RETURN WORK	2006;4	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: AUG 27, 1987
1004.2,2011	WHEN/WHAT CAN EMP PARTIALLY DO	2011;0	1004.67 WORD-PROCESSING
1004.2,2013	20A PROGNOSIS	2000;0	1004.68 WORD-PROCESSING
1004.2,2017	ANY PERMANENT EFFECTS	2006;5	SET ; 'Y' FOR YES ; 'N' LAST EDITED: SEP 28, 1987
1004.2, 2 0 3 0	REASON FOR REFERRAL	2006;6	SET ; 'C' FOR CONSULTATION; 'T' FOR TREATMENT; 'c' FOR CONSULTATION ; 't' FOR TREATMENT LAST EDITED: AUG 21, 1987
1004.2,2040	ICD-9 CODE FOR PRE-EXISTING	2006;7	FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH Input transform: K:SL(X)>61(SL(X)<6) X LAST EDITED: SEP 16, 1987
1004.2,2040.1	ICD-9 CODE FOR DIAGNOSIS	2006;8	FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH Input transform: K:SL(X)>61(SL(X)<6) X LAST EDITED: SEP 16, 1987
1004.2,2201	DUTY STATION AT RECURRENCE	210;4	POINTER TO OACP AGENCY CODES FILE (#1062) LAST EDITED: SEP 23, 1987
1004.2,2201.1	RECURRENCE ADDRESS	210;5	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>541(SL(X)<1) X LAST EDITED: SEP 23, 1987
1004.2,2201.2	RECURRENCE CITY	210;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 23, 1987
1004.2,2201.3	RECURRENCE STATE	210;7	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 23, 1987
1004.2,2201.4	RECURRENCE ZIP CODE	210;8	FREE TEXT FORMAT: NNNN-NNN OR NNNN Input transform: K:SL(X)>101(SL(X)<5)!'(X?SNI(X?SNI"-?"N)) X LAST EDITED: SEP 23, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHINS SYSTEM STORED IN EMPLOY(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,2202	DATE/TIME OF RECURRENCE	210;1	DATE X—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! (Required) Input transform: S %DT="ET" D %DT S X=Y K:Y<1 X LAST EDITED: NOV 2, 1987
1004.2,2202.1	RECURRENCE BASE PAY	214;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1"."2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2202.2	RECURRENCE BASE BASIS	214;2	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2202.3	RECURRENCE SUBSISTENCE	214;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1"."2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2202.35	RECURRENCE SUB BASIS	214;4	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2202.4	RECURRENCE QUARTERS	214;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1"."2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2202.45	RECURRENCE QUART BASIS	214;6	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2202.5	RECURRENCE OTHER RATE	214;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1"."2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2202.55	REC OTHER BASIS	214;8	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2202.6	REC OTHER SPECIFY	214;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 30, 1987
1004.2,2203	REC D/T STOPPED WORK	210;9	DATE Input transform: S %DT="ET" D %DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,2203.1	REC WORK STOPPED BASE RATE	213;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2 K:+X'=X(X'?N1"."2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSRRS/MCHHS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,2203.2	REC WORK STOPPED BASE BASIS	213;2	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2203.3	REC WORK STOPPED SUBSISTENCE	213;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2203.3	SPEC WORK STOPPED SUB BASIS	213;4	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2203.4	REC WORK STOPPED QUARTERS	213;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2203.4	SPEC WKK STOPPED QUART BASIS	213;6	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2203.5	REC WORK STOPPED OTHER RATE	213;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:+X'=X(X'7.NI".2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1004.2,2203.5	SPEC WORK STOPPED OTHER BASIS	213;8	SET; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1004.2,2203.6	REC WORK STOPPED OTHER SPECIFY	213;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1004.2,2204	RECURRENT DT PAY STOPPED	210;10	DATE Input transform: S %DT="ET" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,2205	REC D/T RETURNED TO WORK	210;11	DATE Input transform: S %DT="ET" D %DT S %Y K:Y<1 X LAST EDITED: SEP 30, 1987
1004.2,2206	DID OWP AUTHORIZE TREATMENT	210;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987

DATA DICTIONARY #1004.2 — INJURY COMP AND CLAIMS SUB-FILE OSHRS/ACHHS SYSTEM STORED IN EMPLOY

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1004.2,2207	DATE 1ST TREATMENT AFTER RECLR 210;3		DATE XX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S WDT="E" D ^WDT S X=Y R:Y<1 X LAST EDITED: OCT 25, 1987
1004.2,2209	SUP STATEMENT AT RECLR 211;0		1004.69 WORD-PROCESSING
1004.2,2217	WORK INFO DURING COP/COMP 212;0		1004.705 Multiple
1004.705,.01	DATES & HOURS WORKED 0;1		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:X[""]!(\$A(X)=45) X I \$D(X) R:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1004.705,1	PAY RATE 0;2		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 1, 1987
1004.705,2	TOTAL AMOUNT EARNED 0;3		NUMBER TYPE A NUMBER BETWEEN 0 AND 99999 Input transform: S:X["\$"] X-SP(X,"\$"),2 K:+X'-%L(X'?..N1". "2N)!(X>99999)!(X<0) X LAST EDITED: SEP 1, 1987
1004.705,3	TYPE WORK PERFORMED 0;4		FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 1, 1987
1004.705,4	EMPLOYER NAME 0;5		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 1, 1987
1004.705,5	EMPLOYER STREET ADDRESS 0;6		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 1, 1987
1004.705,6	EMPLOYER CITY/STATE/ZIP 0;7		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 1, 1987

FILES POINTED TO	FIELDS
ACCIDENT (#1025)	TYPE OF ACCIDENT (#112)
ACTIVITIES (#1007)	USUAL WORK ACTIVITY:USUAL WORK ACTIVITY (#.01) ALTERED WORK ACTIVITY:ALTERED WORK ACTIVITY (#.01)
ADJUDICATION STATUS (#1061)	ADJUDICATION STATUS:ADJUDICATION STATUS (#.01)
AGENCY UNIT (#1074)	AGENCY (#11) AGENCY UNIT (#11.5)
AGENT OF ACCIDENT (#1068)	AGENT OF ACCIDENT (#127)
BODY PART (#1018)	BODY PART CODE (#100)
CAUSE OF INJURY (#1026)	CAUSE OF INJURY (#120)
CLAIM TYPE (#1029)	CLAIM TYPE (#202.5)
FORMS (#1002)	CLAIM FORM (#.01) CLAIM FORM LAST RECEIVED (#208) CLAIM FORM NEXT (#209)
NOI (#1017)	NATURE OF INJURY CODE (#101)
OCCUPATION (#1001)	OCCUPATION (#15) OTHER DEPEND OCCUPATION (#540) DEPEND SPOUSE OCCUPATION (#551)
OSHA INJURY CODES (#1057)	INJURY OSHA CODE (#102)
OWCP AGENCY CODES (#1062)	DUTY STATION (#13) DUTY STATION AT REFERENCE (#2201)
OWCP REPORTING OFFICE (#1063)	REPORTING OFFICE (#14)
PAY STATUS (#1093)	PAY STATUS (#21.5)
SOURCE OF INJURY (#1027)	SOURCE OF INJURY (#126)
STATUS (#1003)	STATUS (#9) DUTY STATION STATUS (#13.6) RO STATUS (#14.03) DOCS STATUS (#400.3) RP STATUS (#401.3) STATUS OF SPOUSE (#502.3) OTHER DEPEND STATUS (#504.2) STATUS OF PERSON PAYING (#517.03) STATUS OF EMPLOYER (#548.6) STATUS OF PERSON SIGNING CASB (#555.2) ATTORNEY STATUS (#609.3) STATUS OF NEXT OF KIN (#611.3)

VA STATE (#710.4)
OTHER ANNUITY STATE (#714.6)
TRAINING STATE (#811.3)
MED CARE STATE (#1604.3)
RECURRENCE STATE (#2201.3)
WITNESS INFORMATION:STATE (#3)
THIRD PARTY:STATE (#3)
CHILDREN OF THIS MARRIAGE:STATE (#5)
CHILDREN PRIOR MARRIAGE:STATE (#5)
CHILD & LEGAL GUARDIAN:GUARDIAN'S STATE (#4)
OTHER DEPENDENTS:STATE (#5)
DEPENDENT INFO:STATE (#11)
COMMISSION & SELF-EMP:STATE (#1.3)

STRESSOR (#1083)

STRESSOR INVOLVED INJURY (#132)

DATA DICTIONARY #1005 — ICC FILE OSHRS/ACHHS SYSTEM

STORED IN 'ICC'

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY INJURY/ILLNESS DATE & TIME

'IDENTIFIED' BY NAME

'IDENTIFIED' BY NATURE OF INJURY CODE

'POINTED TO' BY THE 'CASE SERIAL NUMBER' FIELD OF THE 'COP DATA' FILE

'POINTED TO' BY THE 'CASE SERIAL NUMBER' FIELD OF THE 'COMPENSATION PAY' FILE

CROSS-REFERENCED BY: OCCUPATION(AC), TYPE OF ACCIDENT(AD), SOURCE OF INJURY(AE), NATURE OF INJURY CODE(AF),
INJURY/ILLNESS DATE & TIME(AG), NAME(C), INJURY/ILLNESS DATE & TIME(D), TYPE OF PAY RECEIVED(F1),
DATE IMPLEMENTED(F2)

1005,.01 CASE SERIAL NUMBER 0;1

NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 999999999
(Required)
Input transform:

DATA DICTIONARY #1005 — ICC FILE OSHRS/ACHINS SYSTEM

STORED IN ^ICC^

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,13.7	DUTY STATION ZIP CODE	9;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1005,13.9	OSHA SITE CODE	9;10	FREE TEXT ANSWER MUST BE 1-9 CHARACTERS IN LENGTH Input transform: K:\$L(X)>9!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,14	REPORTING OFFICE	10;1	POINTER TO OACP REPORTING OFFICE FILE (#1063) LAST EDITED: SEP 15, 1987
1005,14.01	RO ADDRESS	10;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 17, 1987
1005,14.02	RO CITY	10;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,14.03	RO STATE	10;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1005,14.04	RO ZIP CODE	10;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1005,14.05	POINT OF CONTACT	10;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,14.055	POC TITLE	10;10	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,14.06	POC PHONE	10;7	FREE TEXT ANSWER MUST BE 10-13 CHARACTERS IN LENGTH Input transform: K:\$L(X)>13!(\$L(X)<10) X LAST EDITED: SEP 15, 1987
1005,14.07	POC EXTENSION	10;8	FREE TEXT ANSWER MUST BE 1-4 CHARACTERS IN LENGTH Input transform: K:\$L(X)>4!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,14.08	POC AUTOMON	10;9	FREE TEXT ANSWER MUST BE 7-13 CHARACTERS IN LENGTH Input transform: K:\$L(X)>13!(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1005,14.5	EMPLOYEE COST CENTER	2;4	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH Input transform: K:\$L(X)>6!(\$L(X)<1) X
1005,15	OCCUPATION	3;1	POINTER TO OCCUPATION FILE (#1001) (Required) LAST EDITED: AUG 28, 1987 CROSS-REFERENCE: 1005^AC^M^P^E^S 1)= I \$D(X),X^="" S:\$D(^DIZ(1001,X,0)) \$P(^ICC(DA,3),U,2)-\$P(^DIZ(1001,X,0),U,1) 2)= S \$P(^ICC(DA,3),U,2)-""

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,24	FEGLI CODE	3;12	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!((SL(X)<1) X
1005,25	FEHB CODE	3;13	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!((SL(X)<1) X
1005,26	PRIMARY SUPERVISOR	3;14	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!((SL(X)<1) X
1005,27	SECONDARY SUPERVISOR	3;15	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!((SL(X)<1) X
1005,30	PREMIUM PAY	12;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] N-SP(X,"\$"),2) K:+X'-RL(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1005,31	PREMIUM PAY BASIS	12;2	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1005,32	NIGHT PAY	12;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] N-SP(X,"\$"),2) K:+X'-RL(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1005,33	NIGHT PAY BASIS	12;4	SET ; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1005,34	SUNDAY PAY	12;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] N-SP(X,"\$"),2) K:+X'-RL(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1005,35	SUNDAY PAY BASIS	12;6	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1005,36	OTHER PAY	12;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] N-SP(X,"\$"),2) K:+X'-RL(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1005,37	OTHER PAY BASIS	12;8	SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 17, 1987
1005,38	OTHER PAY SPECIFY	12;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!((SL(X)<1) X LAST EDITED: OCT 17, 1987
1005,50	WITNESS NAME	4;0	1005.01 Multiple

DATA DICTIONARY #1005 — IOC FILE OSHA/ACHIMS SYSTEM

STORED IN ^IOC(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005.01,.01	WITNESS NAME	0;	FREE TEXT ANSWER MUST BE 3-25 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:\$L(X)>25!(\$L(X)<3) X LAST EDITED: SEP 1, 1987
1005.01,1	STREET ADDRESS	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 17, 1987
1005.01,2	CITY	0;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: AUG 17, 1987
1005.01,3	STATE	0;4	POINTER TO STATE FILE (#1003) LAST EDITED: AUG 17, 1987
1005.01,4	ZIP CODE	0;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: AUG 17, 1987
1005.01,5	DATE SIGNED	0;6	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 26, 1987
1005.01,6	WITNESS STATEMENT	1;0	1005.21 WORD-PROCESSING
1005,52	NARRATIVE	6;0	1005.03 WORD-PROCESSING
1005,53	CORRECTIVE ACTION	7;0	1005.04 WORD-PROCESSING
1005,54	REMARKS	8;0	1005.05 WORD-PROCESSING
1005,54.5	SAFETY REMARKS	11;0	1005.07 WORD-PROCESSING
1005,100	BODY PART CODE	100;1	POINTER TO BODY PART FILE (#1018) (Required)
1005,101	NATURE OF INJURY CODE	100;2	POINTER TO NOI FILE (#1017) LAST EDITED: AUG 28, 1987 CROSS-REFERENCE: 1005^AF^MMP5 1)= I \$D(X),X=" " S:\$D(^TU(1017,X,0)) \$P(^IOC(DA,100),U,18)=\$P(^TU(1017,X,0),U,1) 2)= S \$P(^IOC(DA,100),U,18)=" "
1005,101.1	NATURE OF INJURY EXPLANATION	101;0	1005.712 WORD-PROCESSING
1005,101.2	OSHA USE-NOI CODE	100;18	FREE TEXT ANSWER MUST BE 1-5 CHARACTERS IN LENGTH Input transform: K:\$L(X)>5!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005,102	INJURY OSHA CODE	100;3	POINTER TO OSHA INJURY CODES FILE (#1057)
1005,103	DATE OF DEATH	100;4	DATE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X I \$D(X) K:X\$P(^IOC(DA,0),U,2) X LAST EDITED: SEP 14, 1987

DATA DICTIONARY #1005 -- ICC FILE OSHRS/ACHINS SYSTEM

STORED IN 'ICC'

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,104	LOST TIME ACCIDENT	100;5	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,105	LOST WORK DAYS	100;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 999 Input transform: K:+X'-K1(D999)!(X<0)!(X?.E1)". "IN.N) X
1005,106	LIGHT DUTY DAYS	100;7	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 999 Input transform: K:+X'-K1(D999)!(X<0)!(X?.E1)". "IN.N) X
1005,107	NONFATAL LOST WORKDAYS	100;8	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,108	FIRST AID CASE	100;9	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,109	RECORDABLE CASE	100;10	SET ; 'Y' FOR YES ; 'N' FOR NO ; 'R' FOR REMOVED
1005,110	RECORDABLE CASE	100;11	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,111	TERMINATION TRANSFER	100;12	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,112	TYPE OF ACCIDENT	100;13	POINTER TO ACCIDENT FILE (#1025) LAST EDITED: AUG 28, 1987 CROSS-REFERENCE: 1005*AD*MLPFS 1)= I \$D(X),X'="" S:\$D("INV(1027,X,0)) SP("ICIDA,100),U,17)-\$P("INV(1027,X,0),U,1) 2)= S \$P("ICC(DA,100),U,17)=""
1005,112.5	TYPE CODE	100;17	FREE TEXT ANSWER MUST BE 1-5 CHARACTERS IN LENGTH Input transform: K:\$L(X)>5!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005,113	BLDG/SHIP	100;14	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:\$L(X)>15!(\$L(X)<1) X
1005,114	SPECIFIC LOCATION	100;15	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:\$L(X)>40!(\$L(X)<1) X
1005,115	EVOLUTION OF INJURY	100;16	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X
1005,116	EXTENT OF INJURY	110;1	SET; '1' FOR EQUALITY; '2' FOR PERM TOTAL DISABILITY; '3' FOR PERM PARTIAL DISABILITY; '4' FOR NO DISABILITY LIKELY; '5' FOR MISSING LAST EDITED: DEC 23, 1987
1005,117	WORK OPERATION	110;2	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:\$L(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,118	SHIPS STATUS AT INJURY	110;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,119	DAYS HOSPITALIZED	110;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 999 Input transform: K:+X'-K1(D999)!(X<0)!(X?.E1)". "IN.N) X LAST EDITED: SEP 8, 1987

DATA DICTIONARY #1005 -- ICC FILE OSHRS/ACHMS SYSTEM

STORED IN ^ICC

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,120	CAUSE OF INJURY	110;5	POINTER TO CAUSE OF INJURY FILE (#1026) LAST EDITED: AUG 14, 1987
1005,120.1	CAUSE OF INJURY EXPLANATION	111;0	1005.713 WORD-PROCESSING
1005,121	WHO CAUSE OF ACCIDENT	110;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,122	FAILURE TO DO	110;7	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,123	WHY FAILURE	110;8	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,124	DUTY STATUS	120;1	SET (Required); 'ON' FOR ON DUTY; 'OFF' FOR OFF DUTY LAST EDITED: AUG 14, 1987
1005,125	DATE SPEEDLETTER SENT	120;2	DATE XXXX-CAN'T BE ALTERED EXCEPT BY PROGRAMMER Input transform: S \$DT="E" D ^DT S X-Y K:Y<1 X I \$D(X) K:X<SP(^ICC(DA,0),U,2) X LAST EDITED: SEP 23, 1987
1005,126	SOURCE OF INJURY	120;3	POINTER TO SOURCE OF INJURY FILE (#1027) LAST EDITED: AUG 26, 1987 CROSS-REFERENCE: 1005^AE^M^P^S 1)- I \$D(X),X'="" S:\$D(^INJ(1027,X,0)) SP(^ICC(DA,120),U,9)-SP(^INJ(1027,X,0),U,1) 2)- S SP(^ICC(DA,120),U,9)=""
1005,126.5	SOURCE CODE	120;9	FREE TEXT ANSWER MUST BE 1-6 CHARACTERS IN LENGTH Input transform: K:\$L(X)>6!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005,127	AGENT OF ACCIDENT	120;4	POINTER TO AGENT OF ACCIDENT FILE (#1068) LAST EDITED: AUG 14, 1987
1005,128	AGT OF ACC PARTICULAR	120;5	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,129	UNSAFE PRACTICE	120;6	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:\$L(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,130	UNSAFE CONDITION	120;7	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:\$L(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,131	PRE INVOLVED INJURY	120;8	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:\$L(X)>40!(\$L(X)<1) X LAST EDITED: AUG 14, 1987

DATA DICTIONARY #1005 -- IOC FILE OSHRS/NOHMS SYSTEM

STORED IN ~IOC(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,132	STRESSOR INVOLVED INJURY	130;1	POINTER TO STRESSOR FILE (#1063) LAST EDITED: AUG 14, 1987
1005,133	TDY INJURY	130;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,134	RECREATION INJURY	130;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,135	VEHICLE INJURY	130;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,136	TWO WHEEL VEHICLE INJURY	130;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,137	OWCP FILE NUMBER	130;6	FREE TEXT ANSWER MUST BE 1-13 CHARACTERS IN LENGTH Input transform: K:\$(X)>13!(\$\$(X)<1) X LAST EDITED: AUG 14, 1987
1005,138	JOB TRANSFER MED REASONS	130;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,139	RETIRED ON DISABILITY	130;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,140	THIRD PARTY CASE	130;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 14, 1987
1005,141	CASE OUTCOME	130;10	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$(X)>60!(\$\$(X)<1) X LAST EDITED: AUG 14, 1987
1005,141.1	THIRD PARTY	140;0	1005.714(Add New Entry without Asking) Multiple
1005.714,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$(X)>30!(\$\$(X)<1) X LAST EDITED: SEP 1, 1987
1005.714,1	ADDRESS	0;2	FREE TEXT ANSWER MUST BE 1-70 CHARACTERS IN LENGTH Input transform: K:\$(X)>70!(\$\$(X)<1) X LAST EDITED: AUG 26, 1987
1005.714,2	CITY	0;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$(X)>20!(\$\$(X)<1) X LAST EDITED: AUG 26, 1987
1005.714,3	SINCE	0;4	POINTER TO SINCE FILE (#1003) LAST EDITED: AUG 26, 1987
1005.714,4	ZIPCODE	0;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$(X)>10!(\$\$(X)<5) X LAST EDITED: AUG 26, 1987
1005.714,5	SETTLEMENT AMOUNT	0;6	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X("\$" X-SP(X,"\$.2) K:+X'-JL(X'7.NL".2N)!(X-9999999.99)!(X<0) X LAST EDITED: SEP 8, 1988

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,142	DATE TRANSFER FEES TO OACP	130;11	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 14, 1987
1005,143	CLAIMS CLERK ID	130;12	FREE TEXT ANSWER MUST BE 1-3 CHARACTERS IN LENGTH Input transform: K:\$L(X)>3!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,144	INVESTIGATED BY	130;13	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 14, 1987
1005,150	TICKLER FILE	150;0	1005.02 Multiple
1005.02,.01	TICKLER FILE	0;1	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 999999 (Multiply asked) Input transform: ?:+X"=X!(X>999999)!(X<1)!(X?.E1"."IN.N) X LAST EDITED: SEP 25, 1987
1005.02,1	DATE TO DO	0;2	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 25, 1987
1005.02,2	RESPONSIBLE TO DO	0;3	FREE TEXT ENTER PERSON'S INITIALS Input transform: K:\$L(X)>3!(\$L(X)<1) X LAST EDITED: SEP 25, 1987
1005.02,3	ACTION TO TAKE	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: SEP 25, 1987
1005.02,4	REVERSE DATE MMPS CODE: S Y(1005.02,4,1)~SS(\$D(^ICC(D0,150,D1,0)):"(0),1:"") S X=9999999-SP(Y(1005.02,4,1),U,2)		COMPUTED DATE ALGORITHM: 9999999-DATE TO DO
1005,200	WHERE FILED	200;1	SET (Required) ; 'OFF' FOR OFFICIAL PERSONNEL FOLDER ; 'OACP' FOR OFFICE OF WORKER COMPENSATION PROGRAM LAST EDITED: AUG 14, 1987
1005,201	CLAIM RETURN CHECK DATE	200;2	DATE The date must be after the Date of Injury! Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 14, 1987
1005,202	DATE OF CLAIM	200;3	DATE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: SEP 28, 1987
1005,202.5	CLAIM TYPE	200;12	FOUNDER TO CLAIM TYPE FILE (#1029) LAST EDITED: DEC 23, 1987
1005,203	SENT TO OACP	200;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: DEC 23, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,204	DATE CLAIM SENT TO OACP	200;5	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 14, 1987
1005,205	CLAIM CONTOVERIED	200;6	SET (Required) ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 26, 1987
1005,205.1	CONTOVERSION EXPLANATION	305;0	1005.09 WORD-PROCESSING
1005,205.2	CONTOV SUSTAINED BY OACP	200;11	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: OCT 31, 1987
1005,206	ADJUDICATION STATUS	13	1005.16FOINIER Multiple 'IDENTIFIED' BY DATE ENTERED
1005.16,.01	ADJUDICATION STATUS	0	FOINIER TO ADJUDICATION STATUS FILE (#1061) (Required) (Multiply asked) LAST EDITED: JAN 12, 1988 CPL: REFERENCE: 1005.16"B 1)= S %IC(DA(1),13,"B",SE(X,1,30),DA)="" 2)= R %IC(DA(1),13,"B",SE(X,1,30),DA)
1005.16,1	DATE ENTERED	0;2	DATE (Required) Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: JAN 12, 1988
1005.16,2	DATE REMOVED	0;3	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: JAN 12, 1988
1005,207	FINAL DISPOSITION	200;8	SET ; 'A' FOR ACTIVE ; 'I' FOR INACTIVE LAST EDITED: AUG 14, 1987
1005,208	CLAIM FORM LAST RECEIVED	200;9	FOINIER TO FORMS FILE (#1002) Input transform: S DIC("S")="I SP^(0),U,4)=1004" D %DIC K DIC S DIO-DIE,%Y K:Y<0 X SELECT A CLAIM FORM LAST EDITED: OCT 27, 1987
1005,209	CLAIM FORM NEXT	200;10	FOINIER TO FORMS FILE (#1002) Input transform: S DIC("S")="I SP^(0),U,4)=1004" D %DIC K DIC S DIO-DIE,%Y K:Y<0 X SELECT CLAIM FORM LAST EDITED: SEP 25, 1987
1005,210	OCP BEGIN DATE	220;1	DATE XXXX--CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! (Required) Input transform: S %DT="ET" D %DT S %Y K:Y<1 X LAST EDITED: SEP 2, 1987
1005,210.1	OCP NOT TO EXCEED DATE	220;2	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: OCT 31, 1987
1005,210.2	OCP HOURS OF DAY	220;3	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'=%I(X>24)!(X<0)!(X>.E1)".IN.N) X LAST EDITED: OCT 31, 1987

DATA DICTIONARY #1005 — ICC FILE FILE OSHRS/NOHHS SYSTEM STORED IN "ICC"

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,210.3	COP RETURN TO WORK DATE 220;4		DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: OCT 31, 1987
1005,210.4	COP RETURN TO WORK HOURS 220;5		NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'=X1(X<24)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: OCT 31, 1987
1005,210.5	COP RECURRENCE FLAG		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: OCT 31, 1987
1005,210.6	COP DAYS USED	220;7	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 9999 Input transform: K:+X'=X1(X<9999)!(X<0)!(X?.E1)".IN.N) X LAST EDITED: OCT 31, 1987
1005,210.7	COP JOB ORDER COST	220;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X?.N1)".2N)!(X<999999.99)!(X<0) X LAST EDITED: OCT 31, 1987
1005,229	D/T STOPPED WORK	221;1	DATE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="E1" D %DT S %Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,229.1	WORK STOPPED BASE RATE 222;1		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X?.N1)".2N)!(X<999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,229.2	WORK STOPPED BASE BASIS 222;		SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,229.3	WORK STOPPED SUBSISTENCE 222;3		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X?.N1)".2N)!(X<999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,229.35	WORK STOPPED SUB BASIS 222;4		SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,229.4	WORK STOPPED QUARTERS 222;5		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X?.N1)".2N)!(X<999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,229.45	WORK STOPPED QUARTERS BASIS 222;6		SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,229.5	WORK STOPPED OTHER RATE 222;7		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] %SP(X,"\$"),2) K:+X'-%(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,229.55	WORK STOPPED OTHER BASIS 222;8		SET: 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,229.6	WORK STOPPED OTHER SPECIFY 222;9		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 30, 1987
1005,230	D/T PAY STOPPED 221;2		DATE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="ET" D %DT S %Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,231	D/T RETURNED TO WORK 221;3		DATE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S %DT="ET" D %DT S %Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,231.1	RETURNED TO WORK BASE RATE 231;1		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] %SP(X,"\$"),2) K:+X'-%(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,231.2	RETURNED TO WORK BASE BASIS 231;2		SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,231.3	RETURN TO WORK SUBSISTENCE 231;3		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] %SP(X,"\$"),2) K:+X'-%(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,231.35	RETURNED TO WORK SUB BASIS 231;4		SET; 'H' FOR HOURLY; 'W' FOR WEEKLY; 'M' FOR MONTHLY; 'B' FOR BI-MONTHLY; 'A' FOR ANNUALLY; 'D' FOR DAILY LAST EDITED: OCT 18, 1987
1005,231.4	RETURN TO WORK QUARTERS 231;5		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] %SP(X,"\$"),2) K:+X'-%(X'?.NL". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,231.45	RETURNED TO WORK OUR BASIS	231;6	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 18, 1987
1005,231.40	RETURNED TO WORK SALARY	231;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,231.55	RETURNED TO WORK OTHER BASIS	231;8	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 18, 1987
1005,231.6	RETURN TO WORK SPECIFY OTHER	231;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1005,232	DATE RATE CHANGED	232;1	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,232.1	CHANGED BASE RATE	232;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,232.2	CHANGED SUBSISTENCE	232;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,232.3	CHANGED QUARTERS	232;4	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,232.4	CHANGED OTHER RATE	232;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987
1005,232.5	CHANGED OTHER SPECIFY	232;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1005,250	CCP RECVD FROM	250;1	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,250.5	CCP RECVD THRU	250;2	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,250.6	COP RECD GROSS	250;16	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X=-%X(X'7.NI". "2N)!(X>9999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1005,250.7	COP RATE	250;17	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X=-%X(X'7.NI". "2N)!(X>999999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1005,250.8	COP RATE BASIS	250;18	SET ; 'H' FOR HALF-Y ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 18, 1987
1005,251	OTHER LEAVE FROM	250;7	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,251.5	OTHER LEAVE THRU	250;8	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,251.6	OTHER LEAVE SPECIFY	250;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1005,252	SICK LEAVE FROM	250;3	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,252.5	SICK LEAVE THRU	250;4	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,253	ANNUAL LEAVE FROM	250;5	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,253.5	ANNUAL LEAVE THRU	250;6	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,254	COP CLAIMED FROM	250;10	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,254.5	COP CLAIMED THRU	250;11	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,255	COMP CLAIM FROM	250;12	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,255.5	COMP CLAIMED THRU	250;13	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,255.6	SICK/ANNUAL LEAVE CLAIMED FROM	250;14	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,255.7	SICK/ANNUAL LEAVE CLAIMED THRU	250;15	DATE Input transform: S %DT="E" D ^%DT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987

DATA DICTIONARY #1005 — IOC FILE FILE OSHRS/NOHMS SYSTEM STORED IN ^IOC(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,256	COMP RECVD FROM	251;1	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1005,256.1	COMP RECVD THRU	251;2	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1005,256.2	COMP RECVD GROSS	251;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X-SP(X,"S"),2) K:+X'->K(X'?.N1". "2N)!(X0999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1005,256.3	COMPENSATION TYPE	251;4	SET ; 'C' FOR COMPENSATION ; 'S' FOR SCHEDULED AWARD LAST EDITED: NOV 2, 1987
1005,256.4	COMP BEGIN HOURS	251;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'->K1(X>24)!(X<0)!(X?.E1". "IN.N) X LAST EDITED: NOV 2, 1987
1005,256.5	COMP END HOURS	251;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 24 Input transform: K:+X'->K1(X>24)!(X<0)!(X?.E1". "IN.N) X LAST EDITED: NOV 2, 1987
1005,256.6	COMP LEAVE STATUS	251;7	SET ; 'S' FOR SICK LEAVE ; 'P' FOR ADMINISTRATIVE ; 'L' FOR UNOP ; 'T' FOR TEMP OFF SITION ; 'A' FOR ANNUAL LEAVE (UNCRITION) LAST EDITED: NOV 2, 1987
1005,256.7	COMP RECVD DATE	251;8	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,256.8	COMP RECURRENCE FLAG	251;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: NOV 2, 1987
1005,257	MISC PAY RECVD FROM	252;1	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1005,257.1	MISC PAY RECVD THRU	252;2	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: OCT 17, 1987
1005,257.2	MISC PAY RECVD GROSS	252;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X-SP(X,"S"),2) K:+X'->K(X'?.N1". "2N)!(X0999999.99)!(X<0) X LAST EDITED: OCT 17, 1987
1005,257.3	MISC RATE	252;4	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X-SP(X,"S"),2) K:+X'->K(X'?.N1". "2N)!(X0999999.99)!(X<0) X LAST EDITED: OCT 18, 1987
1005,257.4	MISC RATE BASIS	252;5	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: OCT 18, 1987
1005,257.5	MISC SPECIFY	252;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: OCT 18, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,300	DEPENDENTS	300;1	SET ; 'S' FOR SPOUSE ; 'C' FOR CHILDREN ; 'O' FOR OTHER ; 'SC' FOR SPOUSE AND CHILDREN ; 'SO' FOR SPOUSE AND OTHER ; 'SCO' FOR SPOUSE, CHILDREN, & OTHER ; 'CO' FOR CHILDREN AND OTHER ; 'N' FOR NONE LAST EDITED: NOV 12, 1987
1005,301	SICK/ANNUAL LEAVE CLAIMED 300;2 SET (Required) ; 'CO' FOR CONTINUATION OF PAY AND COMPENSATION IF DISABILITY CONTINUES BEYOND 45 DAYS ; 'S/A' FOR SICK AND/OR ANNUAL LEAVE		LAST EDITED: NOV 13, 1987
1005,302	REGULAR WORK HOURS	300;3	FREE TEXT ANSWER IN MILITARY TIME. EX: (0700-1500) Input transform: K:SL(X)>141:(SL(X)<6) X LAST EDITED: AUG 20, 1987
1005,303	REGULAR WORK SCHEDULE	300;4	FREE TEXT EXAMPLE: (M-F, or M,TU,W,TH,F) Input transform: K:SL(X)>151:(SL(X)<3) X LAST EDITED: AUG 20, 1987
1005,303.5	REGULAR NUMBER OF HOURS DAILY	300;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 24 Input transform: K:+X'-X!(X>24)!(X<1)!(X?.E1)".1N.N) X LAST EDITED: OCT 31, 1987
1005,303.6	PARTIME WORK HOURS PER DAY	300;7	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 8 Input transform: K:+X'-X!(X>8)!(X<1)!(X?.E1)".1N.N) X LAST EDITED: OCT 31, 1987
1005,304	DATE RECEIVED NOTICE	300;5	DATE XXXX--CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,307	INJURED IN LINE OF DUTY	300;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1005,307.1	EXPLAIN INJURY	306;0	1005.1 WORD-PROCESSING
1005,308	WILLFUL MISCONDUCT	300;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1005,308.1	EXPLAIN WILLFUL MISCONDUCT	301;0	1005.06 WORD-PROCESSING
1005,310	FIRST EXAM DATE	303;1	DATE XXXX--CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,311	DISABLED FOR WORK	303;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1005,312	SUPS FACTS AGREE WITH EMP	303;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,312.1	SUPERVISOR FACTS	304;0	1005.08 WORD-PROCESSING
1005,314	CLAIM FILING CATEGORY	303;4	SET ; '1' FOR NO LOST TIME AND NO MEDICAL EXPENSE ; '2' FOR NO LOST TIME, MEDICAL EXPENSE INCURRED ; '3' FOR LOST COVERED BY LEAVE, LWOP, OR COP LAST EDITED: AUG 20, 1987
1005,315	RELATES TO ILLNESS	307;0	1005.11 WORD-PROCESSING
1005,316	WHY FILED AFTER 30 DAYS	303;0	1005.12 WORD-PROCESSING
1005,317	REASON FOR STATEMENT DELAY	309;0	1005.13 WORD-PROCESSING
1005,318	WHY MEDICAL REPORTS DELAYED	310;0	1005.14 WORD-PROCESSING
1005,319	DATE LAST EXPOSED TO CONDITION	303;5	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1005,320	DESCRIBE NEW WORK ASSIGNMENT	311;0	1005.15 WORD-PROCESSING
1005,321	DATE 1ST REALIZED	303;6	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1005,322	DATE EMP/CLAIMANT SIGNED	303;7	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1005,324	NEW WORK WEEK	303;8	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!((SL(X)<3) X LAST EDITED: AUG 20, 1987
1005,329	DUTY CHANGED DUE TO DISABILITY	303;9	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 20, 1987
1005,329.1	DESCRIBE CHANGED DUTY	314;0	1005.17 WORD-PROCESSING
1005,330	DATE HEALTH DEDUCTION RESUMED	303;10	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1005,331	DATE OPT INS DEDUCTION RESUMED	303;11	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: AUG 20, 1987
1005,332	DATE HEALTH DEDUCT STOPPED	303;12	DATE Input transform: S %DT="E" D %DT S %Y K:Y<1 X LAST EDITED: SEP 4, 1987

DRAA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,333	NEW HEALTH PLAN DATE	303;13	DATE Input transform: S ^DT="E" D ^DT S X-Y K:Y<1 X LAST EDITED: AUG 20, 1987
1005,400	ATTENDING PHYSICIAN	400;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,400.1	DOCS ADDRESS	400;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 17, 1987
1005,400.2	DOCS CITY	400;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,400.3	DOCS STATE	400;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1005,400.4	DOCS ZIPCODE	400;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1005,400.5	DOCS PHONE NUMBER	400;6	FREE TEXT ANSWER MUST BE 7-14 CHARACTERS IN LENGTH Input transform: K:\$L(X)>14!(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1005,400.6	DOES DOCTOR SPECIALIZE	400;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 15, 1987
1005,400.7	SPECIALTY	400;8	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,400.8	TAX ID	400;9	FREE TEXT ANSWER MUST BE 1-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,400.9	DATE SIGNED	400;10	DATE Input transform: S ^DT="E" D ^DT S X-Y K:Y<1 X LAST EDITED: SEP 15, 1987
1005,401	REFERRAL PHYSICIAN	401;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,401.1	RP ADDRESS	401;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>54!(\$L(X)<1) X LAST EDITED: SEP 17, 1987
1005,401.2	RP CITY	401;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,401.3	RP STATE	401;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987

DATA DICTIONARY #1005 -- IOC FILE FILE OHSWS/NOHMS SYSTEM STORED IN "IOC"

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,401.4	RP ZIPCODE	401;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>101(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1005,401.5	RP PHONE NUMBER	401;6	FREE TEXT ANSWER MUST BE 7-14 CHARACTERS IN LENGTH Input transform: K:\$L(X)>141(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1005,401.6	DOES REF DOCTOR SPECIALIZE	401;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 15, 1987
1005,401.7	RP SPECIALTY	401;8	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,401.8	RP TAX ID	401;9	FREE TEXT ANSWER MUST BE 1-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>101(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,401.9	RP DATE SIGNED	401;10	DATE Input transform: S NOT="E" D NOT S N-Y K:Y<1 X LAST EDITED: SEP 15, 1987
1005,402	SUPERVISOR'S NAME	402;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,402.1	SUPERVISOR'S TITLE	402;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,402.2	SUPERVISOR'S PHONE NUMBER	402;3	FREE TEXT ANSWER MUST BE 7-14 CHARACTERS IN LENGTH Input transform: K:\$L(X)>141(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1005,402.3	SUPERVISOR'S AUTOCON	402;4	FREE TEXT ANSWER MUST BE 7-8 CHARACTERS IN LENGTH Input transform: K:\$L(X)>81(\$L(X)<7) X LAST EDITED: SEP 15, 1987
1005,402.4	DATE SUPERVISOR SIGNED	402;5	DATE Input transform: S NOT="E" D NOT S N-Y K:Y<1 X LAST EDITED: SEP 15, 1987
1005,402.6	SUPERVISORY ACC REPORT	402;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: DEC 9, 1987
1005,502	SPOUSE AT DEPTH	501;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,502.1	ADDRESS OF SPOUSE	501;2	FREE TEXT ANSWER MUST BE 1-55 CHARACTERS IN LENGTH Input transform: K:\$L(X)>551(\$L(X)<1) X LAST EDITED: SEP 15, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,502.2	CITY OF SPOUSE	501;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 15, 1987
1005,502.3	STATE OF SPOUSE	501;4	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1005,502.4	ZIPCODE OF SPOUSE	501;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 15, 1987
1005,503	OTHER DEPEND AT DEATH	502;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1005,504	OTHER DEPEND ADDRESS AT DEATH	502;1	FREE TEXT ANSWER MUST BE 1-55 CHARACTERS IN LENGTH Input transform: K:\$L(X)>55!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005,504.1	OTHER DEPEND CITY	502;2	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:\$L(X)>23!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005,504.2	OTHER DEPEND STATE	502;3	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 11, 1987
1005,504.3	OTHER DEPEND ZIP CODE	502;4	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 9, 1987
1005,505	SERVICE NO	500;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1005,506	BWANCH OF SERVICE	500;4	SET ; 'A' FOR ARMY ; 'N' FOR NAVY ; 'M' FOR MARINES ; 'F' FOR AIR FORCE ; 'C' FOR COAST GUARD LAST EDITED: SEP 21, 1987
1005,507	PERIOD OF SERVICE	500;5	FREE TEXT FORMAT: MO/YR TO MO/YR Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 8, 1987
1005,515	TOTAL BURIAL EXPENSE	505;1	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 99999999 Input transform: K:+X'-#1(0099999999)!(X<0)!(X?.E1)". "IN.N) X LAST EDITED: AUG 21, 1987
1005,516	BURIAL EXPENSE PAID BY VA	505;2	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 99999999 Input transform: K:+X'-#1(0099999999)!(X<0)!(X?.E1)". "IN.N) X LAST EDITED: AUG 21, 1987
1005,517	PARTY PAYING FOR BURIAL	506;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,517.01	ADDRESS OF PERSON PAYING	506;2	FREE TEXT ANSWER MUST BE 1-45 CHARACTERS IN LENGTH Input transform: K:SL(X)>45!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,517.02	CITY OF PER PAYING	506;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,517.03	SIRME OF PER PAYING	506;4	FOUNDER TO SIRME FILE (#1003) LAST EDITED: SEP 15, 1987
1005,517.04	ZIP CODE OF PER PAYING	506;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1005,517.1	AMOUNT OTHER PARTY PAID BURIAL	505;3	NUMBER TYPE A NUMBER BETWEEN .01 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$",2) K:+X'-X(X'7.N1"."2N)!(X>9999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987
1005,520	DOCS NARRATIVE OF REL DISEASE	507;0	1005.3 WORD-PROCESSING
1005,521	DOCS TREATMENT/DIAG	508;0	1005.31 WORD-PROCESSING
1005,522	TREATMENT PRIOR TO DEATH	509;0	1005.32 WORD-PROCESSING
1005,523	TREATMENT DATES (PRE-DEATH)	505;6	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,524	DIRECT CAUSE OF DEATH	510;0	1005.33 WORD-PROCESSING
1005,525	CONTRIBUTORY CAUSE OF DEATH	511;0	1005.34 WORD-PROCESSING
1005,526	WAS DEATH DUE TO WORK	505;7	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,526.1	EXPLAIN CAUSAL RELATIONSHIP	512;0	1005.35 WORD-PROCESSING
1005,527	AUTOPSY/AUTOPSY PERFORMED	505;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,530	SPOUSE DOB	501;6	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,531	DATE OF MARRIAGE	515;1	DATE Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987

DATA DICTIONARY #1005 -- ICC FILE FILE OSHRS/ACHINS SYSTEM STORED IN ICC(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,532	LIVING W/SPOUSE AT DEATH	515;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,533	SPOUSE PREVIOUSLY MARRIED	515;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,534	EMP PREVIOUSLY MARRIED	515;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,535	CHILDREN THIS MARRIAGE	516;0	1005.42 Multiple
1005.42,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:\$L(X)>30!(\$L(X)<3) X LAST EDITED: AUG 21, 1987 CROSS-REFERENCE: 1005.42^B 1)= S ^ICC(DA(1),516,"B",SE(X,1,30),DA)="" 2)= K ^ICC(DA(1),516,"B",SE(X,1,30),DA)
1005.42,1	RELATIONSHIP	0;2	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<3) X LAST EDITED: AUG 21, 1987
1005.42,2	DATE OF BIRTH	0;3	DATE Input transform: S ^DT="E" D ^DT S ^Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005.42,3	ADDRESS	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1005.42,4	CITY	0;5	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:\$L(X)>23!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005.42,5	STATE	0;6	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 11, 1987
1005.42,6	ZIP CODE	0;7	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 9, 1987
1005,536	CHILDREN PRIOR MARRIAGE	517;0	1005.43 Multiple
1005.43,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:\$L(X)>30!(\$L(X)<3) X LAST EDITED: AUG 21, 1987 CROSS-REFERENCE: 1005.43^B 1)= S ^ICC(DA(1),517,"B",SE(X,1,30),DA)="" 2)= K ^ICC(DA(1),517,"B",SE(X,1,30),DA)

DATA DICTIONARY #1005 — ICC FILE FILE OSHRS/ALHIMS SYSTEM STORED IN ^ICC^

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005.43,1	RELATIONSHIP	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1005.43,2	DATE OF BIRTH	0;3	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005.43,3	ADDRESS	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1005.43,4	CITY	0;5	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005.43,5	SIRIE	0;6	FOINIER TO SIRIE FILE (#1003) LAST EDITED: SEP 11, 1987
1005.43,6	ZIP CODE	0;7	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 9, 1987
1005,537	CHILD & LEGAL GUARDIAN	518;0	1005.44 Multiple
1005.44,.01	CHILD NAME	0;1	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:\$L(X)>30!(\$L(X)<3) X LAST EDITED: AUG 21, 1987 CROSS-REFERENCE: 1005.44^B 1)= S ^ICC(DA(1),518,"B",SE(X,1,30),DA)="" 2)= K ^ICC(DA(1),518,"B",SE(X,1,30),DA)
1005.44,1	GUARDIAN NAME	0;2	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<3) X LAST EDITED: AUG 21, 1987
1005.44,2	GUARDIAN'S ADDRESS	0;3	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:\$L(X)>60!(\$L(X)<1) X LAST EDITED: AUG 21, 1987
1005.44,3	CITY	0;4	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:\$L(X)>23!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005.44,4	SIRIE	0;5	FOINIER TO SIRIE FILE (#1003) LAST EDITED: SEP 11, 1987
1005.44,5	ZIP CODE	0;6	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 9, 1988

DATA DICTIONARY #1005 — ICC FILE FILE OSIRIS/ACHES SYSTEM STORED IN ICC

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,538	OTHER DEPENDENTS	519;0	1005.45 Multiple
1005.45,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(SL(X)<3) X LAST EDITED: AUG 21, 1987 CROSS-REFERENCE: 1005.45"B 1)= S "ICC(DA(1),519,"B",SE(X,1,30),DA)="" 2)= K "ICC(DA(1),519,"B",SE(X,1,30),DA)
1005.45,1	RELATIONSHIP	0;2	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<3) X LAST EDITED: AUG 21, 1987
1005.45,2	DATE OF BIRTH	0;3	DATE Input transform: S "DT="E" D "DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005.45,3	ADDRESS	0;4	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X LAST EDITED: AUG 21, 1987
1005.45,4	CITY	0;5	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:SL(X)>23!(SL(X)<1) X LAST EDITED: SEP 9, 1987
1005.45,5	STATE	0;6	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 11, 1987
1005.45,6	ZIP CODE	0;7	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 9, 1987
1005,539	OTHER DEPEND DOB	515;5	DATE Input transform: S "DT="E" D "DT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,540	OTHER DEPEND OCCUPATION	515;6	FOUNDER TO OCCUPATION FILE (#1001) LAST EDITED: SEP 11, 1987
1005,541	OTHER DEPEND SSN	515;7	FREE TEXT ANSWER MUST BE 9 CHARACTERS IN LENGTH Input transform: K:SL(X)>9!(SL(X)<9)!(X?N) X LAST EDITED: SEP 11, 1987
1005,542	OTHER DEPEND RELATIONSHIP	515;8	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<3) X LAST EDITED: AUG 21, 1987
1005,543	PARTIAL/TOTAL DEPENDENCY	515;9	SET ; 'T' FOR TOTAL ; 'P' FOR PARTIAL LAST EDITED: SEP 9, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,544	AMT CONTRIB DEP PRIOR TO DEATH	515;10	NUMBER TYPE A NUMBER BETWEEN .01 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987
1005,545	DEP W/EMP 12 MOS PRIOR TO DIH	520;1	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,546	TOTAL ROOM/BOARD EXTRA	520;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>99999)!(X<0) X LAST EDITED: AUG 24, 1987
1005,546.1	ROOM/BOARD BASIS	520;5	SET ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 10, 1987
1005,547	VALUE RM/ED IF NO FIXED AMT	520;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<0) X LAST EDITED: AUG 24, 1987
1005,547.1	VALUE BASIS	520;4	SET ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 10, 1987
1005,548	TYPE OF WORK PERFORMED	523;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,548.1	PERIOD OF EMPLOYMENT	523;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,548.2	MONTHLY PAY RATE	523;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'=X(X'?.N1". "2N)!(X>9999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1005,548.3	NAME OF EMPLOYER	523;4	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,548.4	ADDRESS OF EMPLOYER	523;5	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,548.5	CITY OF EMPLOYER	523;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987

DATA DICTIONARY #1005 -- ICC FILE FILE OSHRS/ACHENS SYSTEM STORED IN ICC(

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,548.6	STATE OF EMPLOYER	523;7	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1005,548.7	ZIP CODE OF EMPLOYER	523;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1005,549	INVESTMENTS	524;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI". "2N)!(X0999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1005,549.1	PENSIONS OF OTHER DEPENDENT	524;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI". "2N)!(X0999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1005,549.2	PERSONS OTHER THAN EMPLOYEE	524;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI". "2N)!(X0999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1005,549.3	OTHER INCOME FOR DEPENDENTS	524;4	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI". "2N)!(X0999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1005,549.4	TOTAL DEPENDENT INCOME	524;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI". "2N)!(X09999999.99)!(X<0) X LAST EDITED: SEP 15, 1987
1005,550	DEPEND SPOUSE DOB	520;6	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,551	DEPEND SPOUSE OCCUPATION	520;7	FOUNDER TO OCCUPATION FILE (#1001) LAST EDITED: SEP 11, 1987
1005,552	DEPEND SPOUSE MONTHLY WAGE	520;8	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: AUG 21, 1987
1005,553	DEPEND INCOME 12 MOS PRIOR	520;9	NUMBER TYPE A NUMBER BETWEEN .01 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI". "2N)!(X0999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,554	DEPENDS PROPERTY	521;0	1005.48 Multiple
1005.48,.01	DEPENDS PROPERTY	0;1	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH Input transform: K:SL(X)>401(SL(X)<1) X LAST EDITED: SEP 1, 1987 CROSS-REFERENCE: 1005.48 B 1)= S "ICC(DA(1),521,"B",SE(X,1,30),DA)="" 2)= K "ICC(DA(1),521,"B",SE(X,1,30),DA)
1005.48,1	DATE ACQUIRED	0;2	FREE TEXT ANSWER MUST BE 1-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<1) X LAST EDITED: SEP 1, 1987
1005.48,2	VALUE	0;3	NUMBER TYPE A NUMBER BETWEEN .01 AND 9999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-X&(X'?NL"."2N)!(X>9999999.99)!(X<.01) X LAST EDITED: AUG 21, 1987
1005,555	ADDRESS OF PERSON SIGNING OASB	522;1	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>601(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,555.1	CITY OF PERSON SIGNING OASB	522;2	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,555.2	STATE OF PERSON SIGNING OASB	522;3	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1005,555.3	ZIPCODE OF PERSON SIGNING OASB	522;4	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 16, 1987
1005,603	CAUSE OF DEATH	600;4	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<3) X LAST EDITED: AUG 21, 1987
1005,605	EMPLOYED IN DUTY 11 MOS	600;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,606	DUTY AFFORDED 11 MOS WORK	600;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,609	ATTORNEY NAME	601;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 15, 1987

DATA DICTIONARY #1005 — ICC FILE FILE OSHRS/ACHMS SYSTEM STORED IN ICC

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,609.1	ATTORNEY ADDRESS	601;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,609.2	ATTORNEY CITY	601;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,609.3	ATTORNEY STATE	601;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1005,609.4	ATTORNEY ZIP CODE	601;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1005,610	CLAIM FILED WORK	600;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,611	NEXT OF KIN	602;1	FREE TEXT ANSWER MUST BE 1-35 CHARACTERS IN LENGTH Input transform: K:SL(X)>35!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,611.1	ADDRESS OF NEXT OF KIN	602;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,611.2	CITY OF NEXT OF KIN	602;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,611.3	STATE OF NEXT OF KIN	602;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 15, 1987
1005,611.4	ZIP CODE OF NEXT OF KIN	602;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 15, 1987
1005,611.5	RELATIONSHIP OF NEXT OF KIN	602;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 15, 1987
1005,612	LOCATION OF DEATH	600;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 4, 1987
1005,701	DIME WAGES LOST FROM	700;7	DIME Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: SEP 9, 1987
1005,701.5	DIMES WAGES LOST THRU	700;9	DIME Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: SEP 9, 1987
1005,702	HRS COMP CLAIMED	700;2	NUMBER TYPE A NUMBER BETWEEN 0 AND 999 Input transform:

K:4K'-41(X0999)!(X:0)!(X2.E1". "3N.N) X
LAST EDITED: AUG 25, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,703	SCHEDULE AWARD	700;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,704	WAS ANY PAY RECEIVED	700;4	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,708	APPLY/RECD VA BENEFITS	700;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,710	VA CLAIM NUMBER	701;1	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,710.1	VA OFFICE WHERE CLAIM FILED	701;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,710.2	VA STREET ADDRESS	701;3	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,710.3	VA CITY	701;4	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,710.4	VA STATE	701;5	FORWARD TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1005,710.5	VA ZIP CODE	701;6	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 16, 1987
1005,710.6	VA NATURE OF DISABILITY	701;7	FREE TEXT ANSWER MUST BE 1-57 CHARACTERS IN LENGTH Input transform: K:SL(X)>57!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,710.7	VA AMOUNT OF PAYMENT	701;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999.99 Input transform: S:X["\$"] X-SP(X,"\$"),2) K:+X'-+X(X'?.N1". "2N)!(X>99999.99)!(X<0) X LAST EDITED: SEP 16, 1987
1005,713	ANY OTHER RETIRE/DISABILITY	700;10	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,714	OTHER ANNUITIES CLAIM NUMBER	707;1	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,714.1	OTHER ANNUITY DATE BEGAN	707;2	DATE Input transform: S NOT="E" D NOT S X-Y K:Y<1 X LAST EDITED: SEP 16, 1987

1005,714.2 OTHER ANNUITY AMOUNT 707;3

NUMBER TYPE A NUMBER BETWEEN 0 AND 9999999.99

Input transform: S:X "\$" N-SP(X,"\$",2)

K:+X'-N(X'7.NI"."2N)!(X09999999.99)!(X0) X

LAST EDITED: SEP 16, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,714.3	OTHER ANNUITY OFFICE	707;4	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,714.4	OTHER ANNUITY STREET ADDRESS	707;5	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,714.5	OTHER ANNUITY CITY	707;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,714.6	OTHER ANNUITY STATE	707;7	FORWARD TO SOURCE FILE (#1003) LAST EDITED: SEP 16, 1987
1005,714.7	OTHER ANNUITY ZIP CODE	707;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>10!(SL(X)<5) X LAST EDITED: SEP 16, 1987
1005,720	OFFICIAL REMARKS	402	FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:SL(X)>50!(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,722	DEPENDENT INFO	706;0	1005.708 ANSWER MUST BE 1-30 CHARACTERS IN LENGTH LAST EDITED: AUG 21, 1987 Multiple
1005.708,.01	NAME	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH (Multiply asked) Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: AUG 25, 1987 CROSS-REFERENCE: 1005.708"B 1)= S "ICC(DA(1),706,"B",SE(X,1,30),DA)="" 2)= K "ICC(DA(1),706,"B",SE(X,1,30),DA)
1005.708,1	BIRTHDATE	0;2	DATE Input transform: S "DD"="E" D "DD" S X=Y K:Y<1 X LAST EDITED: AUG 25, 1987
1005.708,2	RELATIONSHIP	0;3	FREE TEXT ANSWER MUST BE 1-15 CHARACTERS IN LENGTH Input transform: K:SL(X)>15!(SL(X)<1) X LAST EDITED: AUG 25, 1987
1005.708,3	LIVING WITH YOU	0;4	SET ; 'YES' FOR YES ; 'NO' FOR NO LAST EDITED: SEP 16, 1987
1005.708,5	SUPPORT PAYMENTS?	0;6	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 16, 1987
1005.708,6	ORDERED BY THE COURT?	0;7	SET ; 'Y' FOR YES ; 'N' FOR NO
1005.708,7	AMOUNT OF PAYMENTS	0;8	NUMBER TYPE A NUMBER BETWEEN 0 AND 9999.99 Input transform: S:X["\$"] X-SP(X,"\$",2) K:+X'-X(X?."NI"."2N)!(X>9999.99)!(X<0) X LAST EDITED: SEP 9, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005.708,8	PER	0;9	SET ; 'M' FOR MONTHLY ; 'W' FOR WEEKLY ; 'B' FOR BI-WEEKLY ; 'A' FOR ANNUALLY LAST EDITED: SEP 9, 1987
1005.708,9	STREET ADDRESS	0;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005.708,10	CITY	0;10	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 9, 1987
1005.708,11	STATE	0;11	POINTER TO STATE FILE (#1003)
1005.708,12	ZIP CODE	0;12	FREE TEXT FORMAT: XXXXX-XXXX OR XXXXX Input transform: K:\$L(X)>10!(\$L(X)<5) X LAST EDITED: SEP 9, 1987
1005,727	TOTAL YRS CIV SERVICE	704;2	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 50 Input transform: K:+X'-X!(X>50)! (X<1)! (X?.E1)". "IN.N) X LAST EDITED: SEP 9, 1987
1005,727.5	TOTAL MOS CIV SERVICE	700;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 11 Input transform: K:+X'-X!(X>11)! (X<1)! (X?.E1)". "IN.N) X LAST EDITED: SEP 9, 1987
1005,728	EMP ENROLLED IN FEHB	704;3	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,728.1	PAY PERIOD END DATE FOR HES	700;6	DATE Input transform: S NOT="E" D NOT S X=Y K:Y<1 X LAST EDITED: SEP 9, 1987
1005,729	EMP ENROLLED IN OPT LIFE INS	704;4	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,729.1	LIFE INS OPTION	704;5	SET ; 'A' FOR A ; 'B' FOR B ; 'C' FOR C LAST EDITED: SEP 28, 1987
1005,729.2	HOW MANY OPT/B MULTIPLES	704;6	FREE TEXT ANSWER MUST BE 3-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<3) X
1005,731	PERSON TO CONTACT	700;1	SET ; 'S' FOR SUPERVISOR ; 'O' FOR OTHER LAST EDITED: SEP 16, 1987
1005,801	REFURCHASE LEAVE	704;1	SET ; 'Y' FOR YES ; 'N' FOR NO
1005,802	STATE REASON FOR NOT WORKING	801;0	1005.702 WORD-PROCESSING
1005,806	PAY REC'D DURING COMP	704;7	SET ; 'Y' FOR YES ; 'N' FOR NO

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,808	COMMISSION & SELF-EMP	612;0	1005.726 Multiple
1005.726,.01	DAYS & HOURS WORKED	0;1	FREE TEXT FORMAT: MO/YR-MO/YR XXX HRS (Multiply asked) Input transform: K:\$L(X)>201(\$L(X)<1) X LAST EDITED: AUG 28, 1987 CROSS-REFERENCE: 1005.726~B 1)= S "ICC(DA(1),812,"B",SE(X,1,30),DA)="" 2)= K "ICC(DA(1),812,"B",SE(X,1,30),DA)
1005.726,1	NAME OF BUSINESS	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005.726,1.1	STREET ADDRESS	0;6	FREE TEXT ANSWER MUST BE 1-45 CHARACTERS IN LENGTH Input transform: K:\$L(X)>451(\$L(X)<1) X LAST EDITED: SEP 8, 1987
1005.726,1.2	CITY	0;7	FREE TEXT ANSWER MUST BE 1-23 CHARACTERS IN LENGTH Input transform: K:\$L(X)>231(\$L(X)<1) X LAST EDITED: SEP 8, 1987
1005.726,1.3	STATE	0;8	FOUNDER TO STATE FILE (#1003)
1005.726,1.4	ZIP CODE	0;9	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:\$L(X)>101(\$L(X)<5) X
1005.726,2	SELF-EMPLOYED OR COMMISSION	0;3	SET ; 'S' FOR SELF-EMPLOYED ; 'C' FOR COMMISSION
1005.726,3	TYPE OF ACTIVITY PERFORMED	0;4	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>201(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005.726,4	INCOME DERIVED	0;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005,811	TRAINING REGISTRATION NUMBER	804;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>301(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1005,811.1	TRAINING STREET ADDRESS	804;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:\$L(X)>541(\$L(X)<1) X LAST EDITED: SEP 16, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,811.2	TRAINING CITY	804;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,811.3	TRAINING STATE	804;4	FOUNDER TO STATE FILE (#1003)
1005,811.4	TRAINING ZIP CODE	804;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 16, 1987
1005,811.5	TRAINING DATE OF REGISTRATION	804;6	DATE Input transform: S 'MT'-'E' D 'MT S X-Y K:Y<1 ; LAST EDITED: SEP 16, 1987
1005,812	EMP RECEIVED MONIES FROM ONCP	704;8	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: AUG 21, 1987
1005,816	EXPLAIN CHGS IN EMP HLTH/LIFE	805;0	1005.705 WORD-PROCESSING
1005,1601	TREATMENT/EXAM FOR CAUSE	1600;1	SET ; 'A' FOR AGREE TO FEE SCHEDULE ; 'B' FOR FURNISH NECESSARY TREATMENT NOT SURGERY ; 'C' FOR EXAM FOR JOB RELATED CAUSE & TREATMENT LAST EDITED: AUG 21, 1987
1005,1604	MEDICAL CARE FACILITY	1601;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>301(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,1604.1	MED CARE STREET ADDRESS	1601;2	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>541(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,1604.2	MED CARE CITY	1601;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>201(SL(X)<1) X LAST EDITED: SEP 16, 1987
1005,1604.3	MED CARE STATE	1601;4	FOUNDER TO STATE FILE (#1003) LAST EDITED: SEP 16, 1987
1005,1604.4	MED CARE ZIP CODE	1601;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH Input transform: K:SL(X)>101(SL(X)<5) X LAST EDITED: SEP 16, 1987
1005,1605	HISTORY EMPLOYEE GAVE DOCTOR	1602;0	1005.25 WORD-PROCESSING
1005,1606	PRES/PRIOR CONDITION	1600;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1606.1	EXPLAIN INJ/DIS/IMP	1607;0	1005.26 WORD-PROCESSING
1005,1607	DOE BELIEVE INJ WORK-REL	1600;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,1607.1	DOC EXPLAIN-NOT WORK RELATED 1604;0		1005.27 WORD-PROCESSING
1005,1608	REQUIRE HOSPITALIZATION 1600;4		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1608.1	ADMISSION DATE 1600;6		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1608.2	DISCHARGE DATE 1600;7		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1609	ADDITIONAL HOSPITALIZATION 1600;8		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1610	SURGERY REQUIRED 1600;9		FREE TEXT ANSWER MUST BE 2-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<2) X LAST EDITED: AUG 21, 1987
1005,1611	SURGERY DATE 1600;10		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1612	OTHER TREATMENT PROVIDED 1605;0		1005.28 WORD-PROCESSING
1005,1613	PERMANENT EFFECTS 1606;0		1005.29 WORD-PROCESSING
1005,1614	DATE OF 1ST EXAM 1600;11		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1615	TREATMENT DATES 1607;0		1005.37DATE Multiple
1005.37,.01	TREATMENT DATES 0;1		DATE (Multiply asked) Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1616	DATE DISCHARGED FROM TREATMENT 1600;12		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1617	DATE RESUME LIGHT WORK 1600;13		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1618	DATE RESUME REG WORK 1600;14		DATE Input transform: S ^NOT="E" D ^NOT S ^N-Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1619	EXPLAIN EMP WORK LIMITATIONS 1608;0		1005.39 WORD-PROCESSING
1005,1623	AUTHORIZING OFFICIAL NAME 1610;1		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 16, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,1623.1	AUTHORIZING OFFICIAL TITLE	1610;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 16, 1987
1005,1701	DATE OF EXAM	1700;14	DATE Input transform: S NOT="E" D "NOT S N=Y K:Y<1 X LAST EDITED: SEP 9, 1987
1005,1702	JOB REQUIRES DRIVING	1700;1	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1702.1	SPECIFY VEHICLE	1700;9	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005,1703	JOB REQ OPERATING MACHINERY	1700;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1703.1	SPECIFY MACHINERY	1700;10	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: AUG 28, 1987
1005,1704	HRS PER DAY EMP WORKS	1700;3	NUMBER TYPE A NUMBER BETWEEN 1 AND 8 Input transform: K:+X'=X!(X>8)!(X<1)!(X?.E1)". "N.N) X LAST EDITED: AUG 21, 1987
1005,1705	DAYS PER WEEK EMP WORKS	1700;	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 5 Input transform: K:+X'=X!(X>5)!(X<1)!(X?.E1)". "N.N) X LAST EDITED: AUG 21, 1987
1005,1706	DOES DOC AGREE W/ EMP STATEMENT	1700;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1706.1	EXPLAIN WHAT EMP TOLD YOU	1701;0	1005.46 WORD-PROCESSING
1005,1707	DESCRIBE CLINICAL FINDING	1702;0	1005.47 WORD-PROCESSING
1005,1708	DIAGNOSIS (INJURY)	1703;0	1005.49 WORD-PROCESSING
1005,1709	DIAGNOSIS (OTHER COND)	1704;0	1005.51 WORD-PROCESSING
1005,1710	CAN EMP RETURN TO REG WORK	1700;6	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1710.1	FULL OR PART-TIME	1700;7	SET ; 'F' FOR FULL TIME ; 'P' FOR PART TIME ; 'f' FOR FULL TIME ; 'p' FOR PART TIME LAST EDITED: AUG 21, 1987
1005,1710.11	PART-TIME HRS PER DAY	1700;16	NUMBER TYPE A WHOLE NUMBER BETWEEN 1 AND 8 Input transform: K:+X'=X!(X>8)!(X<1)!(X?.E1)". "N.N) X LAST EDITED: AUG 21, 1987

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,1712	DESCRIBE FURTHER RESTRICTIONS	1706:0	1005.53 WORD-PROCESSING
1005,1713	TOTAL DISABILITY FROM	1707;1	DPDE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: SEP 16, 1987
1005,1713.1	TOTAL DISABILITY TO	1707;2	DPDE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: SEP 16, 1987
1005,1714	PARTIAL DISABILITY FROM	1707;3	DPDE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: SEP 16, 1987
1005,1714.1	PARTIAL DISABILITY TO	1707;4	DPDE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: SEP 16, 1987
1005,1715	ADVISED EMP CAN RESUME WORK	1700;11	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,1715.1	DATE OF ADVICE	1700;12	DPDE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1716	NEXT EXAM DATE	1700;13	DPDE Input transform: S %DT="E" D ^%DT S %Y K:Y<1 X LAST EDITED: AUG 21, 1987
1005,1720	USUAL WORK ACTIVITY	1709;0	1005.18FOUNDER Multiple
1005.18,.01	USUAL WORK ACTIVITY	0;1	FOUNDER TO ACTIVITIES FILE (#1007) (Multiply asked) LAST EDITED: SEP 9, 1987 CROSS-REFERENCE: 1005.18^B 1)= S ^ICC(DA(1),1709,"B",SE(X,1,30),DA)="" 2)= K ^ICC(DA(1),1709,"B",SE(X,1,30),DA)
1005.18,1	DESCRIPTION OF ACTIVITY	0;2	FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:\$L(X)>50!(\$L(X)<1) X LAST EDITED: NOV 12, 1987
1005.18,2	CNT/INTER	0;3	SET ; 'C' FOR CONTINUOUS ; 'I' FOR INTERMITTENT LAST EDITED: SEP 9, 1987
1005.18,3	HOURS PER DAY	0;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 10 Input transform: K:+X'=N1(X>10)!(X<0)!(X?.E1)".1N.N) X LAST EDITED: SEP 9, 1987
1005.18,4	DEGREES F	0;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 200 Input transform: K:+X'=N1(X>200)!(X<0)!(X?.E1)".1N.N) X LAST EDITED: SEP 9, 1987
1005.18,5	DEA	0;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 150 Input transform: K:+X'=N1(X>150)!(X<0)!(X?.E1)".1N.N) X LAST EDITED: SEP 9, 1987

DATA DICTIONARY #1005 -- IOC FILE CSHRS/ACHMS SYSTEM

STORED IN "IOC"

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,1721	ALTERED WORK ACTIVITY	1710;0	1005.2POINTER Multiple
1005.2,.01	ALTERED WORK ACTIVITY	0;1	POINTER TO ACTIVITIES FILE (#1007) (Multiply asked) LAST EDITED: SEP 9, 1987 CROSS-REFERENCE: 1005.2`B 1)= S "IOC(DA(1),1710,"B",SE(X,1,30),DA)=-" 2)= K "IOC(DA(1),1710,"B",SE(X,1,30),DA)
1005.2,1	DESCRIBE ACTIVITY	0;2	FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH Input transform: K:SL(X)>50((SL(X)<1) X LAST EDITED: NOV 12, 1987
1005.2,2	CONT/INTER	0;3	SET ; 'C' FOR CONTINUOUS ; 'I' FOR INTERMITTENT LAST EDITED: SEP 9, 1987
1005.2,3	HOURS PER DAY	0;4	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 10 Input transform: K:+X'-K((X>10))((X<0))((X>.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1005.2,4	DEGREES F	0;5	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 200 Input transform: K:+X'-K((X>200))((X<0))((X>.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1005.2,5	DEA	0;6	NUMBER TYPE A WHOLE NUMBER BETWEEN 0 AND 150 Input transform: K:+X'-K((X>150))((X<0))((X>.E1)".IN.N) X LAST EDITED: SEP 9, 1987
1005,2003	DATE OF MOST RECENT EXAM	2006;1	DATE Input transform: S %DT="E" D "MT S X=Y K:Y<1 X LAST EDITED: AUG 27, 1987
1005,2004	EMP TOTALLY DISABLED FOR WORK	2006;2	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,2005	DESCRIBE PRESENT IMPAIRMENT	2007;0	1005.715 WORD-PROCESSING
1005,2006	20A DIAGNOSIS	2008;0	1005.716 WORD-PROCESSING
1005,2007	TREATMENT/FREQUENCY OF ILL/INJ	2009;0	1005.717 WORD-PROCESSING
1005,2009	UNRELATED DISABILITIES	2010;0	1005.718 WORD-PROCESSING
1005,2010	WILL DISABIL GO EAST 90 DAYS	2006;3	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987

DATA DICTIONARY #1005 -- ICC FILE CSHRS/NOHMS SYSTEM

STORED IN 'ICC'

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,2010.1	EXPECTED LATE RETURN WORK	2006;4	DATE Input transform: S ^DT="E" D ^DT S X=Y K:Y<1 X LAST EDITED: AUG 27, 1987
1005,2011	WHEN/WHAT CAN EMP PARTIALLY DO	2011;0	1005.719 WORD-PROCESSING
1005,2013	20A PROGNOSIS	2000;0	1005.67 WORD-PROCESSING
1005,2017	ANY PERMANENT EFFECTS	2006;5	SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,2030	REASON FOR REFERRAL	2006;6	SET ; 'C' FOR CONSULTATION ; 'T' FOR TREATMENT ; 'c' FOR CONSULTATION ; 't' FOR TREATMENT LAST EDITED: AUG 21, 1987
1005,2040	ICD-9 CODE FOR PRE-EXISTING	2006;7	FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH Input transform: K:SL(X)>6!(SL(X)<6) X LAST EDITED: SEP 16, 1987
1005,2040.1	ICD-9 CODE FOR DIAGNOSIS	2006;8	FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH Input transform: K:SL(X)>6!(SL(X)<6) X LAST EDITED: SEP 16, 1987
1005,2201	DUTY STATION AT RECURRENCE	210;4	POINTER TO OMC AGENCY CODES FILE (#1062) LAST EDITED: SEP 23, 1987
1005,2201.1	RECURRENCE ADDRESS	210;5	FREE TEXT ANSWER MUST BE 1-54 CHARACTERS IN LENGTH Input transform: K:SL(X)>54!(SL(X)<1) X LAST EDITED: SEP 23, 1987
1005,2201.2	RECURRENCE CITY	210;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:SL(X)>20!(SL(X)<1) X LAST EDITED: SEP 23, 1987
1005,2201.3	RECURRENCE STATE	210;7	POINTER TO STATE FILE (#1003) LAST EDITED: SEP 23, 1987
1005,2201.4	RECURRENCE ZIP CODE	210;8	FREE TEXT FORMAT: NNNN-NNN OR N:NNN Input transform: K:SL(X)>10!(SL(X)<5)!(X?N1(X?N1"-")N)) X LAST EDITED: SEP 23, 1987
1005,2202	DATE/TIME OF RECURRENCE	210;1	DATE XXXX--CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! (Required) Input transform: S ^DT="ET" D ^DT S X=Y K:Y<1 X LAST EDITED: NOV 2, 1987
1005,2202.1	RECURRENCE BASE PAY	214;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X("S" X=SP(X,"S",2) K:+X'->X(X?7.N1"."2N)!(X>999999.99)!(X<0) X LAST EDITED: SEP 30, 1987

DATA DICTIONARY #1005 — ICC FILE OSHRS/ACHINS SYSTEM

STORED IN 'ICC'

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005,2202.2	REFERENCE BASE BASIS	214;2	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2202.3	REFERENCE SUBSISTENCE	214;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI".2N)!(X<0) X LAST EDITED: SEP 30, 1987
1005,2202.35	REFERENCE SUB BASIS	214;4	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2202.4	REFERENCE QUARTERS	214;5	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI".2N)!(X<0) X LAST EDITED: SEP 30, 1987
1005,2202.45	REFERENCE QUART BASIS	214;6	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2202.5	REFERENCE OTHER RATE	214;7	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI".2N)!(X<0) X LAST EDITED: SEP 30, 1987
1005,2202.55	REC OTHER BASIS	214;8	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2202.6	REC OTHER SPECIFY	214;9	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: SEP 30, 1987
1005,2203	REC D/T STOPPED WORK	210;9	DATE Input transform: S NOT="ET" D NOT S X=Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,2203.1	REC WORK STOPPED BASE RATE	213;1	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X=SP(X,"\$"),2) K:+X'-XG(X'7.NI".2N)!(X<0) X LAST EDITED: SEP 30, 1987
1005,2203.2	REC WORK STOPPED BASE BASIS	213;2	SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2203.3	REC WORK STOPPED SUBSISTENCE	213;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99

Input transform: S: X["\$" X=SP(X, "\$", 2)
K: +X' -X(X' 7 .NI". "2N)!(X>999999.99)!(X<0) X
LAST EDITED: SEP 30, 1987

DRIA DICTIONARY \$1005 — ICC FILE OSHRS/NCIMS SYSTEM

STORED IN 'ICC'

DRIA ELEMENT	NME	GLOBAL LOCATION	DESCRIPTION
1005,2203.35	REC WORK STOPPED SUB BASIS 213;4		SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2203.4	REC WORK STOPPED QUARTERS 213;5		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X-SP(X,"\$"),2) K:+X'-X(X'7.NI". "2N)1(X0999999.99)1(X0) X LAST EDITED: SEP 30, 1987
1005,2203.45	REC WRK STOPPED QUNT BASIS 213;6		SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2203.5	REC WORK STOPPED OTHER FRIE 213;7		NUMBER TYPE A NUMBER BETWEEN 0 AND 999999.99 Input transform: S:X["\$"] X-SP(X,"\$"),2) K:+X'-X(X'7.NI". "2N)1(X0999999.99)1(X0) X LAST EDITED: SEP 30, 1987
1005,2203.55	REC WORK STOPPED OTHER BASIS 213;8		SET ; 'H' FOR HOURLY ; 'W' FOR WEEKLY ; 'M' FOR MONTHLY ; 'B' FOR BI-MONTHLY ; 'A' FOR ANNUALLY ; 'D' FOR DAILY LAST EDITED: SEP 30, 1987
1005,2203.6	REC WORK STOPPED OTHER SPECIFY 213;9		FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30! (SL(X)<1) X LAST EDITED: SEP 30, 1987
1005,2204	RECURRENT DT PAY STOPPED 210;10		DRIE Input transform: S WDT="ET" D "WDT S X-Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,2205	REC D/T RETURNED TO WORK 210;11		DRIE Input transform: S WDT="ET" D "WDT S X-Y K:Y<1 X LAST EDITED: SEP 30, 1987
1005,2206	DID OACP AUTHORIZE TREATMENT 210;2		SET ; 'Y' FOR YES ; 'N' FOR NO LAST EDITED: SEP 28, 1987
1005,2207	DATE 1ST TREATMENT AFTER RECLR 210;3		DRIE XXXX—CAN'T BE ALTERED EXCEPT BY PROGRAMMER The date must be after the Date of Injury! Input transform: S WDT="E" D "WDT S X-Y K:Y<1 X LAST EDITED: OCT 25, 1987
1005,2209	SUPR STIPENDMENT AT RECLR 211;0		1005.6 WORD-PROCESSING
1005,2217	WORK INFO DURING COP/COMP 212;0		1005.36 Multiple

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
1005.36,.01	DATES & HOURS WORKED	0;1	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:X[""]!((\$A(X)=45) X I \$(X) K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 15, 1987 CROSS-REFERENCE: 1005.36`B 1)= S "ICC(DA(1),212,"B",\$(X,1,30),DA)="" 2)= K "ICC(DA(1),212,"B",\$(X,1,30),DA)
1005.36,1	PAY RATE	0;2	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 1, 1987
1005.36,2	TOTAL AMOUNT EARNED	0;3	NUMBER TYPE A NUMBER BETWEEN 0 AND 99999 Input transform: S:X["\$"] X-SP(X,"\$",2) K:X=X-RL(X"?NL"."2N")!(X>99999)!(X<0) X LAST EDITED: SEP 1, 1987
1005.36,3	TYPE WORK PERFORMED	0;4	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH Input transform: K:\$L(X)>20!(\$L(X)<1) X LAST EDITED: SEP 1, 1987
1005.36,4	EMPLOYER NAME	0;5	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 1, 1987
1005.36,5	EMPLOYER STREET ADDRESS	0;6	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 1, 1987
1005.36,6	EMPLOYER CITY/STATE/ZIP	0;7	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:\$L(X)>30!(\$L(X)<1) X LAST EDITED: SEP 1, 1987

DATA DICTIONARY #1005 — IOC FILE OSHA/ACHHS SYSTEM

STORED IN IOC(

FILES POINTED TO

FIELDS

ACCIDENT (#1025)

TYPE OF ACCIDENT (#112)

ACTIVITIES (#1007)

USUAL WORK ACTIVITY:USUAL WORK ACTIVITY (#.01)
ALTERED WORK ACTIVITY:ALTERED WORK ACTIVITY (#.0

ADJUDICATION STATUS (#1061)

ADJUDICATION STATUS-ADJUDICATION STATUS (#.01)

AGENCY UNIT (#1074)

AGENCY (#11), AGENCY UNIT (#11.5)

AGENT OF ACCIDENT (#1068)

AGENT OF ACCIDENT (#127)

BODY PART (#1018)

BODY PART CODE (#100)

CAUSE OF INJURY (#1026)

CAUSE OF INJURY (#120)

CLAIM TYPE (#1029)

CLAIM TYPE (#202.5)

EMPLOYEE (#1004)

NAME (#2)

FORMS (#1002)

CLAIM FORM LAST RECEIVED (#208)
CLAIM FORM NEXT (#209)

NOI (#1017)

NATURE OF INJURY CODE (#101)

OCCUPATION (#1001)

OCCUPATION (#15)
OTHER DEPEND OCCUPATION (#540)
DEPEND SPOUSE OCCUPATION (#551)

OSHA INJURY CODES (#1057)

INJURY OSHA CODE (#102)

OACF AGENCY CODES (#1062)

DUTY STATION (#13)
DUTY STATION AT RECURRENCE (#2201)

OACF REPORTING OFFICE (#1063)

REPORTING OFFICE (#14)

PAY STATUS (#1093)

PAY STATUS (#21.5)

SOURCE OF INJURY (#1027)

SOURCE OF INJURY (#126)

STATE (#1003)

STATE (#9)
DUTY STATION STATE (#13.6)
FO STATE (#14.03)
DOCS STATE (#400.3)
RP STATE (#401.3)
STATE OF SPOUSE (#502.3)
OTHER DEPEND STATE (#504.2)
STATE OF PER PAYING (#517.03)
STATE OF EMPLOYER (#548.6)
STATE OF PERSON SIGNING CASB (#555.2)
ATTORNEY STATE (#609.3)
STATE OF NEXT OF KIN (#611.3)
VA STATE (#710.4)
OTHER ANNUITY STATE (#714.6)
TRAINING STATE (#811.3)
MED CARE STATE (#1604.3)
REFERENCE STATE (#2201.3)
WITNESS NAME:STATE (#3)
CHILDREN THIS MARRIAGE:STATE (#5)

MED CARE STATE (#1604.3)
 RECURRENCE STATE (#2201.3)
 WITNESS NAME:STATE (#3)
 CHILDREN THIS MARRIAGE:STATE (#5)
 CHILDREN PRIOR MARRIAGE:STATE (#5)
 CHILD & LEGAL GUARDIAN:STATE (#4)
 OTHER DEPENDENTS:STATE (#5)
 DEPENDENT INFO:STATE (#11)
 THIRD PARTY:STATE (#3)
 COMMISSION & SELF-EMP:STATE (#1.3)

STRESSOR (#1083)

STRESSOR INVOLVED INQUIRY (#132)

INPUT TEMPLATE(S): (NONE)

PRINT TEMPLATE(S):

ADDITIONITION STATUS	JAN 13,1988 USER #0	ADDITIONITION STATUS REPORT
ALL INQUIRIES REPORT	OCT 1,1987 USER #0	ALL INQUIRIES
CAPTIONED	USER #0	
CASE RANGE SUMMARY	FEB 18,1988 USER #0	CASE SUMMARY BY DATE/NAME
CASE SUMMARY REPORT	FEB 18,1988 USER #0	CASE SUMMARY
CRNAV 5100/10 LOG	OCT 5,1987 USER #0	OSH LOG
PAY INFORMATION	NOV 3,1987 USER #0	PAY INFORMATION BY TYPE OF PAY RECEIVED
PERSONAL CASE SUMMARY	OCT 1,1987 USER #0	PERSONNEL INQUIRY/ILLNESS REPORT
PRINT INQUIRY LOG	SEP 28,1987 USER #0	INQUIRY LOG REPORT
SPEEDLETER/2	MAR 17,1988 USER #0	
TICKLER REPORT2	JAN 20,1988 USER #0	TICKLER REPORT

SORT TEMPLATE(S):

NI	SEP 30,1987 USER #0	
NUMBER @		
PAY BEGIN DATE	NOV 3,1987 USER #0	DATE IMPLEMENTED @ (User is asked range)
SELECT BY DATE IMPLEMENTED		
T3CSAMS	DEC 8,1987 USER #0	NAME PAY INFORMATION
SORTS PAY INFORMATION BY SOURCE		
T3CSORT1 DUTY STA/SHOP/DATE	OCT 16,1987 USER #0	' Print Template always used
DUTY STATION "13 (User is asked range)		
SHOP+" (User is asked range)		
DATE(INQUIRY/ILLNESS DATE & TIME)" (User is asked range)		
USER SELECT BY DUTY STATION, SHOP, AND INQUIRY DATE		
T3CSORT2 EMPLOYEE	NOV 13,1987 USER #0	" Print Template always used
NAME "2;@; (User is asked range) Selection by Employee		
NOTE: Do NOT use the same employee as first and last selection!		
T3CSORT3 DUTY STA/CASE/DATE	OCT 19,1987 USER #0	" Print Template always used
DUTY STATION "13; (User is asked range)		
CASE SERIAL NUMBER'; (User is asked range)		
DATE(INQUIRY/ILLNESS DATE & TIME)"'; (User is asked range)		
USER SELECT BY DUTY STATION, CASE SERIAL NUMBER, AND INQUIRY DATE		
T3CSORT4 DUTY STA/DATE	OCT 19,1987 USER #0	
DUTY STATION "13; (User is asked range)		
DATE(INQUIRY/ILLNESS DATE & TIME)"'; (User is asked range)		
USER SELECT BY DUTY STATION AND INQUIRY DATE		
T3CSORT5 DATE/NAME	OCT 19,1987 USER #0	
DATE(INQUIRY/ILLNESS DATE & TIME))"'"; (User is asked range)		
NAME'"2; (User is asked range)		
USER SELECT BY INQUIRY DATE AND EMPLOYEE NAME		

TSCSORT6 CASE/COP BEGIN DATE OCT 19,1987 USER #0
CASE SERIAL NUMBER)@'; (User is asked range)
DATE(COP BEGIN DATE)''; (User is asked range)
USER SELECT BY CASE SERIAL NUMBER AND DATE COP BEGAN

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

This file consists of the physical activities that an injured person can do after an injury. This table is used in forms (CA-17, CA-18).

'IDENTIFIED' BY ACTIVITY/EXPOSURE

'POINTED TO' BY THE 'USUAL WORK ACTIVITY' FIELD OF THE 'USUAL WORK ACTIVITY' SUB-FIELD
OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD
OF THE 'EMPLOYEE' FILE

'POINTED TO' BY THE 'ALTERED WORK ACTIVITY' FIELD OF THE 'ALTERED WORK ACTIVITY' SUB-FIELD
OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD
OF THE 'EMPLOYEE' FILE

'POINTED TO' BY THE 'USUAL WORK ACTIVITY' FIELD OF THE 'USUAL WORK ACTIVITY' SUB-FIELD
OF THE 'ICC FILE' FILE

'POINTED TO' BY THE 'ALTERED WORK ACTIVITY' FIELD OF THE 'ALTERED WORK ACTIVITY' SUB-FIELD
OF THE 'ICC FILE' FILE

CROSS-REFERENCED BY: ACTIVITY CODE(B), ACTIVITY/EXPOSURE(C)

CREATED SEP 9, 1987

1007,.01	ACTIVITY CODE	0;1	<p>FREE TEXT ANSWER MUST BE 1 CHARACTER IN LENGTH (Required) Input transform: K:\$L(X)>1!(\$L(X)<1) X CROSS-REFERENCE: 1007"B 1)= S INU(1007,"B",SE(X,1,30),DA)="" 2)= K INU(1007,"B",SE(X,1,30),DA)</p>
1007,1	ACTIVITY/EXPOSURE	0;2	<p>FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:X!""""!(\$A(X)=45) X I \$D(X) K:\$L(X)>60!(\$L(X)<1) X CROSS-REFERENCE: 1007"C 1)= S INU(1007,"C",SE(X,1,30),DA)="" 2)= K INU(1007,"C",SE(X,1,30),DA)</p>

DATA DICTIONARY #1017 — NATURE OF INJURY (NOI) FILE OSHRS/ACHMS SYSTEM STORED IN INJ(1017, (44 ENTRIES)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY NATURE OF INJURY

'FOINDED TO' BY THE 'NATURE OF INJURY CODE' FIELD OF THE 'IOC FILE' FILE

'FOINDED TO' BY THE 'NATURE OF INJURY' FIELD OF THE 'INJURY' FILE

'FOINDED TO' BY THE 'SECONDARY NATURE OF INJURY' FIELD

OF THE 'SECONDARY BODY PART' SUB-FIELD OF THE 'INJURY' FILE

'FOINDED TO' BY THE 'NOI PER EMPLOYEE' FIELD OF THE 'CLAIM' SUB-FIELD

DATA DICTIONARY #1017 — NATURE OF INJURY (NOI) FILE OSHRS/ACHMS SYSTEM STORED IN INJ(1017, (44 ENTRIES)
OF THE 'INJURY' FILE

'FOINDED TO' BY THE 'NATURE OF INJURY' FIELD OF THE 'INJ LOG UPDATE' FILE

CROSS-REFERENCED BY: NOI CODE(B), NATURE OF INJURY(C), NATURE OF INJURY(D)

1017,.01	NOI CODE	0;1	FREE TEXT ANSWER MUST BE 2-3 CHARACTERS IN LENGTH (Required) NATURE OF INJURY CODE
----------	----------	-----	---

Input transform:

K:SL(X)>3!(SL(X)<2)!('X'71P.E)!(X'7.ANP) X

LAST EDITED: AUG 6, 1985

Source of data: TABLE NATIONALLY DEFINED

GROUP: INJ DIZ(1017, INDIVIDUAL COPIES

CROSS-REFERENCE: 1017'B

1)= S INJ(1017,"B",SE(X,1,30),DA)=""

2)= K INJ(1017,"B",SE(X,1,30),DA)

1017,1	NATURE OF INJURY	0;2	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH (Required) Input transform:
--------	------------------	-----	---

K:X!""!(SA(X)=45) X I SD(X) K:SL(X)>20!(SL(X)<1) X

LAST EDITED: FEB 18, 1986

CROSS-REFERENCE: 1017'C'WIC

1)= F %1:1 Q:SP(X," ",%,99)="" S I-SP(X," ",%)

I SL(I)>2,SL(I)<31,LD("WIC")'[I S INJ(1017,"C",I,DA)=""

2)= F %1:1 Q:SP(X," ",%,99)="" S I-SP(X," ",%)

I SL(I)>2,SL(I)<31 K INJ(1017,"C",I,DA)

CROSS-REFERENCE: 1017'D

1)= S INJ(1017,"D",SE(X,1,30),DA)=""

2)= K INJ(1017,"D",SE(X,1,30),DA)

DATA DICTIONARY #1018 -- BODY PART FILE OSHRS/ACHIPS SYSTEM STORED IN INJ(1018, (82 ENTRIES)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY BODY PART DESCRIPTION
 'POINTED TO' BY THE 'BODY PART CODE' FIELD OF THE 'IOC FILE' FILE
 'POINTED TO' BY THE 'BODY PART' FIELD OF THE 'INJURY' FILE
 'POINTED TO' BY THE 'SECONDARY BODY PART' FIELD OF THE 'SECONDARY BODY PART' SUB-FIELD
 OF THE 'INJURY' FILE
 'POINTED TO' BY THE 'BODY PART INJ PER EMPL' FIELD OF THE 'CLAIM' SUB-FIELD
 OF THE 'INJURY' FILE
 'POINTED TO' BY THE 'BODY PART' FIELD OF THE 'INJ LOG UPDATE' FILE
 CROSS-REFERENCED BY: BODY PART CODE(B), BODY PART DESCRIPTION(C), BODY PART DESCRIPTION(D)

1018,.01	BODY PART CODE	0;1	<p>FREE TEXT ANSWER MUST BE 2-3 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>3!(SL(X)<2)!(X'7LP.E)!(X'?.ANP) X LAST EDITED: AUG 6, 1985 Source of data: TABLE NATIONALLY DEFINED GROUP: INJ DIZ(1018, INDIVIDUAL COPIES ANATOMICAL LOCATION OF INJURY CROSS-REFERENCE: 1018"B 1)= S INJ(1018,"B",SE(X,1,30),DA)="" 2)= K INJ(1018,"B",SE(X,1,30),DA)</p>
----------	----------------	-----	---

1018,1	BODY PART DESCRIPTION	0;2	<p>FREE TEXT ANSWER MUST BE 2-30 CHARACTERS IN LENGTH (Required) Input transform: K:X[""]!(SA(X)=45) X I SD(X) K:SL(X)>30!(SL(X)<2) X LAST EDITED: FEB 18, 1986 CROSS-REFERENCE: 1018"C"KWIC 1)= F %1:1 Q:SP(X," ",%,99)="" S I=SP(X," ",%) I SL(I)>2,SL(I)<31,DD"KWIC")!(I S INJ(1018,"C",I,DA)="" 2)= F %1:1 Q:SP(X," ",%,99)="" S I=SP(X," ",%) I SL(I)>2,SL(I)<31 K INJ(1018,"C",I,DA) CROSS-REFERENCE: 1018"D 1)= S INJ(1018,"D",SE(X,1,30),DA)="" 2)= K INJ(1018,"D",SE(X,1,30),DA)</p>
--------	-----------------------	-----	--

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY ACCIDENT TYPE

'FOINIED TO' BY THE 'TYPE OF ACCIDENT' FIELD OF THE 'TOC FILE' FILE

'FOINIED TO' BY THE 'TYPE OF ACCIDENT' FIELD OF THE 'INJURY' FILE

'FOINIED TO' BY THE 'INJURY' TYPE CODE' FIELD OF THE 'CLAIM' SUB-FIELD
OF THE 'INJURY' FILE

CROSS-REFERENCED BY: ACCIDENT ID(B), ACCIDENT TYPE(C)

1025,.01 ACCIDENT ID 0;1

NUMBER TYPE A WHILE NUMBER BETWEEN 0 AND 9999
(Required) ACCIDENT TYPE IDENTIFIER

Input transform:

K:~X'~X!(X>9999)!(X<0)!(X?.E1". "IN.N) X

LAST EDITED: AUG 6, 1985

Source of data: TABLE NATIONALLY DEFINED

GROUP: INJ DIZ("ACC", INDIVIDUAL COPIES

CROSS-REFERENCE: 1025"B

1)= S "INJ(1025,"B",SE(X,1,30),DA)=""

2)= F "INJ(1025,"B",SE(X,1,30),DA)

1025,1 ACCIDENT TYPE 0;2

FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH

(Required) Input transform:

K:X["""1(SA(X)=45) X I SD(X) K:SL(X)>40!(SL(X)<1) X

CROSS-REFERENCE: 1025"C"KWIC

1)= F %1:1 Q:SP(X," ",%,99)="" S I=SP(X," ",%)

I SL(I)>2,SL(I)<31, "ID("KWIC")'(I S "INJ(1025,"C",I,DA)=""

2)= F %1:1 Q:SP(X," ",%,99)="" S I=SP(X," ",%)

I SL(I)>2,SL(I)<31 K "INJ(1025,"C",I,DA)

DATA DICTIONARY #1026 — CAUSE OF INJURY FILE OSHRS/ACHINS SYSTEM STORED IN INJ(1026, (99 ENTRIES)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY DESCRIPTION

'POINTED TO' BY THE 'CAUSE OF INJURY' FIELD OF THE 'ICC FILE' FILE

'POINTED TO' BY THE 'CAUSE OF INJURY' FIELD OF THE 'INJURY' FILE

CROSS-REFERENCED BY: CAUSE ID(B), DESCRIPTION(C)

1026,.01	CAUSE ID	0;1	<p>NUMBER TYPE A WHILE NUMBER BETWEEN 1 AND 99 (Required)</p> <p>Input transform: K:+X'=XI(X>99)!(X<1)!(X?.E1)". "IN.N) X</p> <p>Source of data: TABLE NATIONALLY DEFINED</p> <p>GROUP: INJ DIZ("CAUSE", INDIVIDUAL COPIES</p> <p>CROSS-REFERENCE: 1026~B</p> <p>1)= S "INJ(1026,"B",SE(X,1,30),DA)=""</p> <p>2)= K "INJ(1026,"B",SE(X,1,30),DA)</p>
----------	----------	-----	--

1026,1	DESCRIPTION	0;2	<p>FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH (Required) Input transform:</p> <p>K:X[""""]!(SA(X)=45) X I SD(X) K:SL(X)>50!(SL(X)<1) X</p> <p>CROSS-REFERENCE: 1026~C</p> <p>1)= S "INJ(1026,"C",SE(X,1,30),DA)=""</p> <p>2)= K "INJ(1026,"C",SE(X,1,30),DA)</p>
--------	-------------	-----	--

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY DESCRIPTION

'POINTED TO' BY THE 'SOURCE OF INJURY' FIELD OF THE 'IOC FILE' FILE

'POINTED TO' BY THE 'SOURCE OF INJURY' FIELD OF THE 'INJURY' FILE

'POINTED TO' BY THE 'INJURY SOURCE CODE' FIELD OF THE 'CLAIM' SUB-FIELD

OF THE 'INJURY' FILE

CROSS-REFERENCED BY: SOURCE ID(B), DESCRIPTION(C)

1027,.01	SOURCE ID	0;1
----------	-----------	-----

FREE TEXT ANSWER MUST BE 1-4 CHARACTERS IN LENGTH
(Required) Input transform: K:\$L(X)>4!(\$L(X)<1) X

LAST EDITED: SEP 23, 1987

Source of data: TABLE NATIONALLY DEFINED

GROUP: INJ DIZ("SOI", INDIVIDUAL COPIES

CROSS-REFERENCE: 1027~B

1)= S INJ(1027,"B",SE(X,1,30),DA)=""

2)= K INJ(1027,"B",SE(X,1,30),DA)

1027,1	DESCRIPTION	0;2
--------	-------------	-----

FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH
(Required) Input transform:

K:X[""]!(\$A(X)-45) X I \$D(X) K:\$L(X)>60!(\$L(X)<1) X

CROSS-REFERENCE: 1027~C

1)= S INJ(1027,"C",SE(X,1,30),DA)=""

2)= K INJ(1027,"C",SE(X,1,30),DA)

DRIA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY DESCRIPTION
 'POINTED TO' BY THE 'CLAIM TYPE' FIELD OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD
 OF THE 'EMPLOYEE' FILE
 'POINTED TO' BY THE 'CLAIM TYPE' FIELD OF THE 'IOC FILE' FILE
 'POINTED TO' BY THE 'CLAIM TYPE' FIELD OF THE 'CLAIM' SUB-FIELD
 OF THE 'INJURY' FILE
 CROSS-REFERENCED BY: TYPE(B), DESCRIPTION(C)

1029,.01	TYPE	0;1	<p>FREE TEXT ANSWER MUST BE 2 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>21(SL(X)<2)1'(X'71P.E)1(X'7.ANP) X Source of data: TABLE NATIONALLY DEFINED GROUP: INU DIZ("CLAIMTYPE", INDIVIDUAL COPIES CROSS-REFERENCE: 1029"B 1)- S INU(1029,"B",SE(X,1,30),DA)="" 2)- R INU(1029,"B",SE(X,1,30),DA)</p>
----------	------	-----	---

1029,1	DESCRIPTION	0;2	<p>FREE TEXT ANSWER MUST BE 1-25 CHARACTERS IN LENGTH (Required) Input transform: K:IN*****1(\$A(X)=45) X 1 \$D(X) K:SL(X)>251(SL(X)<1) X LAST EDITED: OCT 10, 1985 CROSS-REFERENCE: 1029"C 1)- S INU(1029,"C",SE(X,1,30),DA)="" 2)- R INU(1029,"C",SE(X,1,30),DA)</p>
--------	-------------	-----	---

CURRENT ENTRIES

TYPE: H1	DESCRIPTION: HEARING LOSS
TYPE: SS	DESCRIPTION: STRAIN/SPRAIN
TYPE: CO	DESCRIPTION: CORUSION/ABRASION
TYPE: FR	DESCRIPTION: FRACTURE/LACERATIONS
TYPE: BU	DESCRIPTION: BURN
TYPE: OI	DESCRIPTION: OTHER
TYPE: DS	DESCRIPTION: DERMATITIS/SKIN
TYPE: LU	DESCRIPTION: GENERAL LUNG DISORDERS
TYPE: AF	DESCRIPTION: ANXIETY/FATIGUE
TYPE: HE	DESCRIPTION: HEMAT
TYPE: PO	DESCRIPTION: EPOXY POISONING

TYPE: CT

DESCRIPTION: CARPAL TUNNEL SYNDROME

TYPE: MS

DESCRIPTION: MISCELLANEOUS ILLNESS

TYPE: AS

DESCRIPTION: ASBESTOSIS

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY INU CODE DESCRIPTION
 'POINTED TO' BY THE 'INJURY OSHA CODE' FIELD OF THE 'ICC FILE' FILE
 'POINTED TO' BY THE 'INJURY OSHA CODE' FIELD OF THE 'INJURY' FILE
 'POINTED TO' BY THE 'INJURY OSHA CODE' FIELD OF THE 'INU LOG UPDATE' FILE
 CROSS-REFERENCED BY: OSHA INJURY CODE(B)

1057,.01	OSHA INJURY CODE	0;1	FREE TEXT ANSWER MUST BE 2 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>2!(SL(X)<2)!'(X?2N) X Source of data: TABLE NATIONALLY DEFINED GROUP: INU DIZ(1057, INDIVIDUAL COPIES CROSS-REFERENCE: 1057~B 1)= S "INU(1057,"B",SE(X,1,30),DA)="" 2)= K "INU(1057,"B",SE(X,1,30),DA)
----------	------------------	-----	---

1057,1	INU CODE DESCRIPTION	0;2	FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH Input transform: K:SL(X)>60!(SL(X)<1) X
--------	----------------------	-----	---

CURRENT ENTRIES

OSHA INJURY CODE: 10	INU CODE DESCRIPTION: ALL OCCUPATIONAL INJURIES
OSHA INJURY CODE: 21	INU CODE DESCRIPTION: OCCUPATIONAL SKIN DISEASES OR DISORDERS
OSHA INJURY CODE: 22	INU CODE DESCRIPTION: DUST DISEASES OF THE LUNGS
OSHA INJURY CODE: 23	INU CODE DESCRIPTION: RESPIRATORY CONDITIONS DUE TO TOXIC AGENTS
OSHA INJURY CODE: 24	INU CODE DESCRIPTION: POISONING (SYSTEMIC EFFECTS OF TOXIC MATERIALS)
OSHA INJURY CODE: 25	INU CODE DESCRIPTION: PHYSICAL AGENT DISORDERS (NOT TOXIC MATERIAL)
OSHA INJURY CODE: 26	INU CODE DESCRIPTION: DISORDERS DUE TO REPORTED TRAUMA
OSHA INJURY CODE: 29	INU CODE DESCRIPTION: ALL OTHER OCCUPATIONAL ILLNESSES
OSHA INJURY CODE: 20	INU CODE DESCRIPTION:
OSHA INJURY CODE: 11	INU CODE DESCRIPTION: Y

DATA DICTIONARY #1061 -- ADDUICATION STATUS FILE OS/RS/ACHMS SYSTEM
ES)

STORED IN INU(1061, (70 ENTRI

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY STATUS DESCRIPTION

'POINTED TO' BY THE 'ADDUICATION STATUS' FIELD OF THE 'ADDUICATION STATUS' SUB-FIELD
OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD
OF THE 'EMPLOYEE' FILE

'POINTED TO' BY THE 'ADDUICATION STATUS' FIELD OF THE 'ADDUICATION STATUS' SUB-FIELD
OF THE 'ICC FILE' FILE

'POINTED TO' BY THE 'ADDUICATION STATUS' FIELD OF THE 'CLAIM' SUB-FIELD
OF THE 'INJURY' FILE

CROSS-REFERENCED BY: ADDUICATION STATUS(B), STATUS DESCRIPTION(C)

1061,.01 ADDUICATION STATUS 0;1

FREE TEXT ANSWER MUST BE 2-3 CHARACTERS IN LENGTH
(Required) Input transform:

K:SL(X)>3!(SL(X)<2)!(X'71P.E)!(X'7.ANF) X

LAST EDITED: JAN 14, 1988

Source of data: TABLE NATIONALLY DEFINED

GROUP: INU DIZ(1061, INDIVIDUAL COPIES

CROSS-REFERENCE: 1061"B

1)= S INU(1061,"B",SE(X,1,30),DA)=""

2)= K INU(1061,"B",SE(X,1,30),DA)

1061,1 STATUS DESCRIPTION 0;2

FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH
(Required) Input transform:

K:X[""]!(SA(X)=45) X I SD(X) K:SL(X)>60!(SL(X)<1) X

LAST EDITED: JUL 9, 1986

CROSS-REFERENCE: 1061"C

1)= S INU(1061,"C",SE(X,1,30),DA)=""

2)= K INU(1061,"C",SE(X,1,30),DA)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
'POINTED TO' BY THE 'DUTY STATION' FIELD OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD OF THE 'EMPLOYEE' FILE 'POINTED TO' BY THE 'DUTY STATION AT REFERENCE' FIELD OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD OF THE 'EMPLOYEE' FILE 'POINTED TO' BY THE 'DUTY STATION' FIELD OF THE 'ICC FILE' FILE 'POINTED TO' BY THE 'DUTY STATION AT REFERENCE' FIELD OF THE 'ICC FILE' FILE 'POINTED TO' BY THE 'INJURY DUTY STATION' FIELD OF THE 'INJURY' FILE 'POINTED TO' BY THE 'DUTY STATION' FIELD OF THE 'CLAIM' SUB-FIELD OF THE 'INJURY' FILE 'POINTED TO' BY THE 'REFERENCE DUTY STATION' FIELD OF THE 'CLAIM' SUB-FIELD OF THE 'INJURY' FILE CROSS-REFERENCED BY: AGENCY DUTY STATION NAME(B)			
1062,.01	AGENCY DUTY STATION NAME	0;1	FREE TEXT ANSWER MUST BE 3-45 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>45!(SL(X)<3)!('X'?1P.E)!('X'?7.ANP) X Source of data: TABLE LOCALLY DEFINED GROUP: INU OACRORG(INDIVIDUAL COPIES CROSS-REFERENCE: 1062'B 1)= S "INU(1062,"B",SE(X,1,30),DA)="" 2)= K "INU(1062,"B",SE(X,1,30),DA)
1062,1	OACP AGENCY CODE	0;2	FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>6!(SL(X)<6) X
1062,2	ACTIVITY UIC	0;3	FREE TEXT ANSWER MUST BE 5-6 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>6!(SL(X)<5) X
1062,3	CLAIMANT CODE	0;4	FREE TEXT ANSWER MUST BE 2-4 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>4!(SL(X)<2) X
1062,4	DUTY STATION ADDRESS	0;5	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>40!(SL(X)<1) X
1062,5	DUTY STATION CITY	0;6	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>20!(SL(X)<1) X
1062,6	DUTY STATION STATE	0;7	POINTER TO STATE FILE (#1003) (Required)
1062,7	DUTY STATION ZIPCODE	0;8	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>10!(SL(X)<5) X
1062,8	REPORTING OFFICE	0;9	POINTER TO OACP REPORTING OFFICE FILE (#1063) (Required) LAST EDITED: OCT 7, 1985
1062,9	PRIOR OACP AGENCY CODE	1;0	1062.01 Multiple
1062.01,.01	PRIOR OACP AGENCY CODE		

0;1

FREE TEXT ANSWER MUST BE 6 CHARACTERS IN LENGTH

Input transform: K:SL(X)>6!(SL(X)<6) X

LAST EDITED: OCT 7, 1985

DATA DICTIONARY #1062 -- OACP AGENCY CODES FILE OSHRS/ACHMS SYSTEM

STORED IN INJ(1062,

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

1062.01,1	LAST DATE CODE ACTIVE	0;2	DATE (Required) Input transform: S %DT="E" D %DT S X=Y K:Y<1 X LAST EDITED: OCT 7, 1985
-----------	-----------------------	-----	---

1062.10	OSHA SITE CODE	0;10	FREE TEXT ANSWER MUST BE 1-9 CHARACTERS IN LENGTH Input transform: K:SL(X)>9!(SL(X)<1) X LAST EDITED: OCT 15, 1985
---------	----------------	------	--

FILES POINTED TO	FIELDS
OACP REPORTING OFFICE (#1063)	REPORTING OFFICE (#6)
STATE (#1003)	DUTY STATION STATE (#6)

PRINT TEMPLATE(S):

OACP AGENCIES	OCT 7, 1985 USER #0	OACP AGENCY CODES LIST
---------------	---------------------	------------------------

DATA ELEMENT	NAME	GLOBL LOCATION	DESCRIPTION
'POINTED TO' BY THE 'REPORTING OFFICE' FIELD OF THE 'INJURY COMP AND CLAIMS' SUB-FIELD OF THE 'EMPLOYEE' FILE 'POINTED TO' BY THE 'REPORTING OFFICE' FIELD OF THE 'IOC FILE' FILE 'POINTED TO' BY THE 'REPORTING OFFICE' FIELD OF THE 'CLAIM' SUB-FIELD OF THE 'INJURY' FILE 'POINTED TO' BY THE 'REPORTING OFFICE' FIELD OF THE 'OACP AGENCY CODES' FILE CROSS-REFERENCED BY: NAME(B)			
1063,.01	NAME	0;1	NAME MUST BE 3-30 CHARACTERS, NOT NUMERIC OR STARTING WITH PUNCTUATION (Required) Input transform: K:SL(X)>30!((X?N)!(SL(X)<3)!'(X'7IP.E)!(X'7.ANP) X Source of data: TABLE LOCALLY DEFINED GROUP: INJ OACPREPT(INDIVIDUAL COPIES CROSS-REFERENCE: 1063"B 1)= S "INJ(1063,"B",SE(X,1,30),DA)="" 2)= K "INJ(1063,"B",SE(X,1,30),DA)
1063,1	REPORTING OFFICE ADDRESS	0;2	FREE TEXT ANSWER MUST BE 1-40 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>40!(SL(X)<1) X
1063,2	REPORTING OFFICE CITY	0;3	FREE TEXT ANSWER MUST BE 1-20 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>20!(SL(X)<1) X
1063,3	REPORTING OFFICE STATE	0;4	POINTER TO STATE FILE (#1003) (Required)
1063,4	REPORTING OFFICE ZIPCODE	0;5	FREE TEXT ANSWER MUST BE 5-10 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>10!(SL(X)<5) X
1063,5	POINT OF CONTACT NAME	0;6	FREE TEXT ENTER NAME IN LAST, FIRST MI FORMAT (Required) Input transform: K:SL(X)>25!(SL(X)<1) X
1063,5.5	POINT OF CONTACT TITLE	0;10	FREE TEXT ANSWER MUST BE 1-30 CHARACTERS IN LENGTH Input transform: K:SL(X)>30!(SL(X)<1) X LAST EDITED: FEB 18, 1986
1063,6	POINT OF CONTACT PHONE	0;7	FREE TEXT ANSWER MUST BE 10-13 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>13!(SL(X)<10) X
1063,7	POINT OF CONTACT EXTENSION	0;8	FREE TEXT ANSWER MUST BE 4 CHARACTERS IN LENGTH Input transform: K:SL(X)>4!(SL(X)<4) X LAST EDITED: OCT 7, 1985
1063,8	POINT OF CONTACT AUTOMOV	0;9	FREE TEXT ANSWER MUST BE 7-13 CHARACTERS IN LENGTH Input transform: K:SL(X)>13!(SL(X)<7) X

LAST EDITED: OCT 7, 1985

FILES POINTED TO
STATE (#1003)

FIELDS
REPORTING OFFICE STATE (#3)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
-----------------	------	--------------------	-------------

'IDENTIFIED' BY AGENCY OF ACCIDENT

'POINTED TO' BY THE 'AGENT OF ACCIDENT' FIELD C. THE 'ICC FILE' FILE

'POINTED TO' BY THE 'AGENCY OF ACCIDENT' FIELD OF THE 'INJURY' FILE

CROSS-REFERENCED BY: AGENT CODE(B), AGENCY OF ACCIDENT(C)

1068,.01	AGENT CODE	0;1	<p>FREE TEXT ANSWER MUST BE 4 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>4!(SL(X)<4)!(X'7LP.E)!(X'7.ANP) X Source of data: TABLE NATIONALLY DEFINED GROUP: INJ DIZ("AGENT", INDIVIDUAL COPIES CROSS-REFERENCE: 1068^b 1)= S INJ(1068,"B",SE(X,1,30),DA)="" 2)= K INJ(1068,"B",SE(X,1,30),DA)</p>
----------	------------	-----	--

1068,1	AGENCY OF ACCIDENT	0;2	<p>FREE TEXT ANSWER MUST BE 1-50 CHARACTERS IN LENGTH (Required) Input transform: K:X("")(SA(X)=45) X I SD(X) R:SL(X)>50!(SL(X)<1) X CROSS-REFERENCE: 1068^c 1)= S INJ(1068,"C",SE(X,1,30),DA)="" 2)= K INJ(1068,"C",SE(X,1,30),DA)</p>
--------	--------------------	-----	---

DATA DICTIONARY #1093 -- PAY STATUS FILE OSHRS/ACHHS SYSTEM

STORED IN "INU(1093, (22 ENTRIES)

DATA ELEMENT	NAME	GLOBAL LOCATION	DESCRIPTION
<p>'IDENTIFIED' BY PAY STATUS 'POINTED TO' BY THE 'PAY STATUS' FIELD OF THE 'ICC FILE' FILE 'POINTED TO' BY THE 'PAY STATUS' FIELD OF THE 'INJURY' FILE CROSS-REFERENCED BY: PAY STATUS CODE(B), PAY STATUS(C)</p>			
1093,.01	PAY STATUS CODE	0;1	<p>FREE TEXT ANSWER MUST BE 2 CHARACTERS IN LENGTH (Required) Input transform: K:SL(X)>2!(SL(X)<2)!(X'?_AMP) X Source of data: TABLE NATIONALLY DEFINED GROUP: INU DIZ(1093, INDIVIDUAL COPIES CROSS-REFERENCE: 1093"B 1)= S "INU(1093,"B",SE(X,1,30),DA)="" 2)= K "INU(1093,"B",SE(X,1,30),DA)</p>
1093,1	PAY STATUS	0;2	<p>FREE TEXT ANSWER MUST BE 1-60 CHARACTERS IN LENGTH (Required) Input transform: K:X["""!(\$A(X)=45) X I \$D(X) K:SL(X)>60!(SL(X)<1) X CROSS-REFERENCE: 1093"C 1)= S "INU(1093,"C",SE(X,1,30),DA)="" 2)= K "INU(1093,"C",SE(X,1,30),DA)</p>

APPENDIX D
ROUTINE DOCUMENTATION

APPENDIX D
ROUTINE DOCUMENTATION

LABEL	FUNCTION
T3CF	Employee CA Print Programs.
T3CF1	Creates a temporary global for CA forms.
T3CF2	Prints out CA form.
T3CFB	Creates a temporary global for blank forms.
T3CFI	Prints out instructions for CA forms.
T3CFSEL	Selected look up for forms.
T3COMP	Stores COMP information in ^INJ(1009.
T3COP	Stores COMP information in ^INJ(1008.
T3COPD	Creates COP not to exceed datel.
T3CPYNF	Stores pay information in pay information file.
T3CFILE	Stores the claim forms into the employee data file.
T3CPAK	Create an ICC package.

```

T3CF ; THIS PROGRAM EMPLOY CA PRINT PROG ; 09 Dec 87 7:13 PM
INIT S IOF="#,*27,*91,*50,*74,*27,*91,*72" W @IOF D ^T3CFSEL
Q: ^SD(FN)!(^SD(E0)) S: FN<1 FN=0 Q: ^SD(^FORMS(FN))
D ^T3CF1,KILL S V=";" ,TCN="," ,U="^" ;SET UP FORM
;
LOOP F %1=1:1 W: ^(%#10) "." S TCN=SO(^FORMS(FN,3,TCN)) Q: TCN="^"(TCN^7.N)
--> S PG=SS(^SD(^FORMS(FN,3,TCN,0)):1,$P(^FORMS(FN,3,TCN,0),U,5)'="":
--> $P(^FORMS(FN,3,TCN,0),U,5),1:1),ST=0
--> F %2=1:1 S ST=SO(^FORMS(FN,3,TCN,2,ST)) Q: ST="^" S GLO1=GLO D GDT
D STORE,MLTI W "]" ,! K: ^FM("B",SJ,2)<4 ^FM(SJ,2),^FM("B",SJ,2)
END D KILL W !,"ENTER DEVICE THE FORM WILL PRINT ON " D ^T3CF2 Q
KILL K TX,TXL,TEXT,SEL,NOD,DF1,A2,A1,PG,TCN,ST,PEC,TB,LN,GLB,%,%2,%3,%4,
--> X1,CNT,GLO1,GLO2,MULT Q
;
;
STORE ;;CREATE ONLY ONE STRING PER LINE
F PG=1,2 S LN="^" F %1=1:1 S LN=SO(^FM(SJ,PG,LN)) Q: LN="^"
--> S (TEXT,TX,BL,TAB)="^" F %1=1:1 S TAB=SO(^FM(SJ,PG,LN,TAB)) Q: TAB="^"
--> S TEXT=^FM(SJ,PG,LN,TAB) D FILL S ^FM(SJ,PG,LN,1)=TX
QUIT ;
FILL S TXL=SL(TX) G: TAB=1 F10
I TXL=TAB!(TXL>TAB) S TX=TX " " G F10
F %99=TXL:1: TAB=1 S TX=TX "-"
F10 S TX=TX TEXT,TXL=SL(TX),CNT=0
F11 I SL(TX)>132 G: SE(TX,SL(TX))' "-" " F12 S TX=SE(TX,1,SL(TX)-1) G F11
F12 I SL(TX)>132 G: SP(TX," ",1)=TX!(CNT>10) F13 S CNT=CNT+1,
--> TX=SP(TX," ",1) " " SP(TX," ",2,999) G F12
F13 K: TAB=1 ^FM(SJ,PG,LN,TAB) Q
;
MLTI ;INSERT MULTIPLE RECORDS INTO THE NUMBER OF SPACES ALLOWED
F PG=1,2 S LN="^" F %1=1:1 S LN=SO(^FM(SJ,PG,LN)) Q: LN="^" I LN["." D M1
Q ;
M1 D MK S LN1=SP(LN,".",1),(BEG,LN2)=LN1,(NA,MM,CU,CNT)=0 F %99=1:1
--> S LN1=SO(^FM(SJ,PG,LN1)) Q: SP(LN1,".",1)'=BEG S NA=NA+1
S END=SP(LN1,".",1)-1 G: LN+1=LN1 M3
M2 S CNT=0 F %99=BEG:1:END Q: CNT>NA I ^SD(^FM(SJ,PG,%99,1))
--> S LN2=SO(^FM(SJ,PG,LN2)),^FM(SJ,PG,%99,1)=^FM(SJ,PG,LN2,1),CNT=CNT+1
F %99=1:1:NA S BEG=SO(^FM(SJ,PG,BEG)) K ^FM(SJ,PG,BEG)
MK K TEXT,LN1,LN2,CNT,NA,BEG,END,MM,CU,%99,%98 Q
M3 F %99=LN:1 Q: ^SD(^FM(SJ,PG,%99)) S MM=MM+1
S CNT=MM-1 F %99=1:1 S LN1=SO(^FM(SJ,PG,LN1)) Q: LN1["."
S CU=0 F %99=CNT:MM Q: (BEG+%99>60)!(%99>LN1) S CU=CU+1
G: CU=0 M3K F %99=1:1:CU Q: %99>NA S LN2=SO(^FM(SJ,PG,LN2))
F %98=LN2:1:LN2+CNT Q: ^SD(^FM(SJ,PG,%98,1))
--> S TEXT(PG,END,1)=^FM(SJ,PG,%98,1),END=END+1
M3K F %99=BEG:1: CNT+BEG F %98=.01:.01:NA K ^FM(SJ,PG,%99,%98)
S TEXT="^" F %99=1:1 S TEXT=SO(TEXT(PG,TEXT)) Q: TEXT="^"
--> S ^FM(SJ,PG,TEXT,1)=TEXT(PG,TEXT,1)
S LN=LN1-1 K TEXT,LN1,LN2,CNT,NA,BEG,END,MM,CU,%99,%98 Q
;
GDT ;GATHER DATA FROM EMPLOYEE FILE
;A1=POINTER,A2=NODE,DF1=DATA DEFINITION,TB=COLUMN,LN=LINE
S
MULT=1,A1=SP(^FORMS(FN,3,TCN,2,ST,0),U,1),A2=SP(^FORMS(FN,3,TCN,2,ST,0),U,4)

```

```

GDT0 S TB=$P(^FORMS(FN,3,TCN,2,ST,0),U,2),LN=$P(^FORMS(FN,3,TCN,2,ST,0),U,3)
Q:TB="!"(LN="")!(A1="")!(PG="") Q:A1=".01&(A2="")
I A1=".01&(A2="") S:'SD(^FM($J,PG,LN,TB))
--> ^FM($J,PG,LN,TB)=$P(^EMPLOY(E0,0),U,1) Q
Q:A2<1004!(A2>1004.999) S:A2>1004.2&(^SD(^DD(A2,A1,0))) A1=.01
Q:'SD(^DD(A2,A1,0))
S DF1=$P(^DD(A2,A1,0),U,2) Q:DF1="" ;DATA DEFINITION
I DF1? NP S A1=.01,A2=DF1 G GDT0
GDT1 I A2>1004.2 D MULT Q:MULT=0
;DATA IS AT THE 1004.2—INJURY COMP/CLAIMS LEVEL
S PEC=$P($P(^DD(A2,A1,0),U,4),V,2),NOD=$P($P(^DD(A2,A1,0),U,4),V,1)
G:DF1["W"] WP ;WORD PROCESSING
S GLO1=GLO1 NOD ") S X=$S(^SD(@GLO1):",1:$P(@GLO1,U,PEC))
Q:X="&(A2=I004.2)
D @S(DF1["S": "SEL",DF1["P": "PTR",DF1["D": "DTE",DF1["F": "FREE",1:"GDT2"])
I DF1["W"&(A2>1004.2) S MULT=MULT+1 G GDT1
GDT2 QUIT ;
FREE Q:X="" D:A2>1004.2 CHECK S ^FM($J,PG,LN,TB)=X K X Q ;FREE TEXT
MULT S GLO1=GLO,GLO2="" I A2=1004.23 S NODE=$P(^FORMS(FN,3,TCN,2,ST,0),U,5)
--> S:NODE="" NODE="ERROR" S GLO1=GLO1 NODE "", " Q
S NAME=$O(^DD(A2,0,"NM",""),UP=^DD(A2,0,"UP"),
S ELEM=$O(^DD(UP,"B",NAME,""),NODE=$P($P(^DD(UP,ELEM,0),U,4),V,1)
S:NODE="" NODE="ERROR" S GLO1=GLO1 NODE "", " Q:DF1["W"
S LN=LN+.01,GLO1=GLO1_MULT,GLO2=GLO1_"") S:'SD(@GLO2) MULT=0
S GLO1=GLO1 "", " Q
DTE Q:X="" D:A2>1004.2 CHECK S X1=$P(X,".",2)
S:X'="" ^FM($J,PG,LN,TB)=SE(X,4,5) "/" SE(X,6,7) "/" SE(X,2,3)
S:X1'="" ^FM($J,PG,LN,TB)=^FM($J,PG,LN,TB) " AT: " X1 K X1,X Q
PTR Q:X="" I X'? N D:A2>1004.2 CHECK S ^FM($J,PG,LN,TB)=X Q
S PEC=$S($P(^FORMS(FN,3,TCN,2,ST,0),U,6)="":1,
--> 1:$P(^FORMS(FN,3,TCN,2,ST,0),U,6)),
--> S GLB="" $P(^DD(A2,A1,0),U,3) X ",0" ")
I ^SD(@GLB) D:A2>1004.2 CHECK S ^FM($J,PG,LN,TB)=$P(@GLB,U,PEC) K X,GLB Q
K X,GLB Q
SEL Q:X="" S SEL=$P(^DD(A2,A1,0),U,3) F %4=1:1 S X1=$P($P(SEL,V,%4),":",1)
Q:X1="" I X1=X D:A2>1004.2 CHECK S ^FM($J,PG,LN,TB)=$P(SEL,V,%4),":",2)
Q
WP ;WORD PROCESSING
S CNT=0 F %3=1:1 S GLO2=GLO1 %3 ",0)" Q:'SD(@GLO2)
Q:$D(^FM($J,PG,LN+CNT,TB)) S ^FM($J,PG,LN+CNT,TB)=@GLO2,CNT=CNT+1
QUIT
CHECK I ^SD(^FM($J,PG,LN,TB)) S LN=LN+.01 G CHECK
Q

```

***** INDEX OF T3CF *****

LOCAL VARIABLES LINE OCCURRENCES

(* Changed ! Killed)

```

8
>> %1 LOOP*,KILL!,STORE+1*,MLTI+1*
%2 STORE+1*
%3 LOOP*,KILL!
%3 KILL!,WP+1*
%4 KILL!,SEL*

```

```

%98      MK!,M3+3*,M3K*,M3K+2!
%99      FILL+2*,M1*,M2*,M2+1*,MK!,M3*,M3+1*,M3+2*,M3+3*,M3K*,M3K+1*
          M3K+2!
A1       KILL!,GDT+2*,GDT0,GDT0+1,GDT0+2*,GDT0+3,GDT0+4*,GDT1+1,PTR+1
          SEL
A2       KILL!,GDT+2*,GDT0,GDT0+1,GDT0+2,GDT0+3,GDT0+4*,GDT1,GDT1+1
          GDT1+3,GDT1+4,FREE,MULT,MULT+1,DTE,PTR,PTR+1,PTR+2,SEL
BEG      M1*,M2,M2+1*,MK!,M3+2,M3K,M3K+2!
>> BL   STORE+1*
CNT      KILL!,FI0*,FI2*,M1*,M2*,MK!,M3+1*,M3+2,M3+3,M3K,M3K+2!,WP+1*
CU       M1*,MK!,M3+2*,M3+3,M3K+2!
DF1      KILL!,GDT0+3*,GDT0+4,GDT1+2,GDT1+4,MULT+1
>> E0    INIT,GDT0+1
>> ELEM  MULT+1*
END      M1+1*,M2,MK!,M3+3*,M3K+2!
>> FN    INIT*,LOOP,GDT+2,GDT+3,MULT,PTR+1
GLB      KILL!,PTR+1*,PTR+2!,PTR+3!
>> GLO   LOOP,MULT
GLO1     LOOP*,KILL!,GDT1+3*,MULT*,MULT+1*,WP+1
GLO2     KILL!,MULT*,MULT+1*,WP+1*
>> IOF   INIT*
LN       KILL!,STORE+1*,FI*,MULT+1*,M1,M1+1,M3,M3K+2*,GDT+3*,GDT0
          GDT0+1,FREE,MULT*,DTE,PTR,PTR+2,SEL,WP+1,CHECK*
LN1      M1*,M1+1,MK!,M3+1*,M3+2,M3K+2!
LN2      M1*,M2*,MK!,M3+3*,M3K+2!
MM       M1*,MK!,M3*,M3+1,M3+2,M3K+2!
MULT     KILL!,GDT+2*,GDT1,GDT1+4*,MULT+1*
NA       M1*,M2,M2+1,MK!,M3+3,M3K,M3K+2!
>> NAME  MULT+1*
NOD      KILL!,GDT1+1*,GDT1+3
>> NOD   MULT*,MULT+1*
PEC      KILL!,GDT1+1*,GDT1+3,PTR+1*,PTR+2
PG       LOOP*,KILL!,STORE+1*,FI3,MULT+1*,M1,M2,M2+1,M3,M3+1,M3+3
          M3K,M3K+1,GDT0,GDT0+1,FREE,DTE,PTR,PTR+2,SEL,WP+1,CHECK
          KILL!,SEL*
SEL      LOOP*,KILL!,GDT+2,GDT+3,MULT,PTR+1
ST       STORE+1*,FILL,FILL+1*,FILL+2,FI3
>> TAB  KILL!,GDT+3*,GDT0,GDT0+1,FREE,DTE,PTR,PTR+2,SEL,WP+1,CHECK
TB       INIT+1*,LOOP*,KILL!,GDT+2,GDT+3,MULT,PTR+1
TCN      KILL!,STORE+1*,FI0,MK!,M3K+1*,M3K+2!
TEXT     M3+3*,M3K+1
>> TEXT( KILL!,STORE+1*,FILL,FILL+1*,FILL+2*,FI0*,FI1*,FI2*
TX       KILL!,FILL*,FILL+1,FILL+2,FI0*
TXL      INIT+1*,LOOP,GDT+2,GDT+3,GDT0+1,GDT0+3,GDT1+1,GDT1+3,MULT
>> U     MULT+1,PTR+1,PTR+2,SEL
UP       MULT+1*
>> V     INIT+1*,GDT+1,MULT+1,SEL
X        GDT1+3*,FREE,DTE!,PTR,PTR+1,PTR+2!,PTR+3!,SEL
XI       KILL!,DTE*,SEL

```

GLOBAL VARIABLES (* Changed 1 Killed)

```

'DO(      GDT0+2,GDT0+3,GDT1+1,MULT+1,PTR+1,SEL
'EMPLOY(   GDT0+1

```

```

^FM(      LOOP+1!,STORE+1*,FI3!,MLTI+1,M1,M2*.M2+1!,M3,M3+1,M3+3,M3K!
          M3K+1*,GDT0+1*,FREE*,DTE*,PTR*,PTR+2*.SEL*,WP+1*,CHECK
^FORMS(    INIT,LOOP,GDT+2,GDT+3,MULT,PTR+1

```

NAKED GLOBALS

NONE

LABEL REFERENCES

@(GDT1+4
CHECK	FREE,DTE,PTR,PTR+2,SEL,CHECK
F10	FILL,FILL+1
F11	F11
F12	F11,F12
F13	F12
FILL	STORE+1
GDT	LOOP
GDT0	GDT0+4
GDT1	GDT1+4
KILL	INIT+1.END
M1	MLTI+1
M3	M1+1
M3K	M3+3
MR	M1
MLTI	LOOP+1
MULT	GDT1
STORE	LOOP+1
WP	GDT1+2

EXTERNAL REFERENCES

^T3CF1	INIT+1
^T3CF2	END
^T3CFSEL	INIT

```

T3CF1 ;THIS PROGRAM CREATES A TEMPORATY GLOEAL FOR CA FORMS;DMP;14 Nov 87
INIT Q:'SD(FN) I 'SD('FORMS(FN,0)) W !,1," INVALID SELECTION " Q
S U="",(TB,PG,LN,LC)="",POS="",N=1,ALPHA="abcdefghijklmnopq",C=0
S (HD,LINE,SP)=" F %99=1:1:132 S LINE=LINE " "
S (P2,SHT)="",(OLN,HCT)=0,LP=60,BP=1,TB=1 K FM
BEG W !,"Compiling forms [...]" d HEADER,DATA
S FM("B",%0,1)=LC(1),FM("B",%J,2)=LC(2),FM(%J,0)=FN
END K CNT,DE,SECN,SEC,I1,I2,I3,I4,I5,ALPHA,C,D1,D2,D3
K C1,C2,C3,C4,LN,OLN,POS,TEXT,TP,X1B,X1E,XE,XE
K I6,N,P2,PG,SHT,TAB,TLN,U,V,D,I,X,HCT,I1
Q
TAB ;CREATES A TAB CHARACTER
S SP="" Q:'SD(TB1) F %99=1:1:TB1 S SP=SP_ " "
S TBX=SP TBX K TB1,SP Q
HEADER F %99=1,2 S FM(%J,%99,BP,1)=SP('FORMS(FN,0),U,1)_ " "
--> SP('0),U,2),LC(%99)=BP,LC=BP
F %1=1,5 S PG=SS(%5:2,1:1),LC(PG)=LC I SD('FORMS(FN,%0)) S HD=0
--> F %1=1:1 S HD=SC('FORMS(FN,%HD)) Q:HD=""
--> S LC(PG)=LC(PG)+1,FM(%J,PG,LC(PG),1)=FORMS(FN,%HD,0)
F %1=1,2 S LC(%1)=LC(%1)+1,FM(%J,%1,LC(%1),1)=LINE,FM(%J,%1,LP+3,1)=LINE.
--> TBX=SP('FORMS(FN,0),U,1) " REVISION DATE: "
--> SP('0),U,3),TB1=132-$L(TBX)\2 D TAB
--> S FM(%J,%1,LP+4,1)=TBX,FM("B",%J,%1)=LC(%1)+1
K TBX,TB1,TAB,%1,%99,% Q
DATA ;CREATES DATA FORMAT
S (CNT,I2)=0,LPG=1
F I=1:1 Q:'SD('FORMS(FN,6,"B",I)) D DE
WRO F I1=1:1 S I2=SN(TCF(I2)) Q:I2=-1 S C3=TCF(I2),C4=I2 D WR1,L2
Q
WR1 S C2=SP('FORMS(FN,3,C3,1,0),"",3),PG=SP('FORMS(FN,3,C3,0),U,5)
S:PG="" PG=LPG S LPG=PG
F I3=1:1 C2 D:SD('FORMS(FN,3,C3,1,I3,0)) F2 Y TCF(C4)
Q
DE S SEC=SO('FORMS(FN,6,"B",I,""),SECN=SP('FORMS(FN,6,SEC,0),"",2)
F I1=1:1 S CNT=SO('FORMS(FN,3,"C",SECN,CNT)) Q:CNT=""
--> S:SD('FORMS(FN,3,CNT,0)) DE=SP('FORMS(FN,3,CNT,0),"",1),
--> TCF(DE=CNT,SECN,I)=SECN
Q
L2 F I4=1:1 S C1=I2_SE(ALPHA,I4) Q:'SD(TCF(C1)) S C3=TCF(C1),C4=C1 D WR1
Q
F2 S TEXT=SP('FORMS(FN,3,C3,1,I3,0),"",1)
S TP=SP('FORMS(FN,3,C3,1,I3,0),"",2)
S LN=SP('FORMS(FN,3,C3,1,I3,0),"",3)
Q:LN=""(TP="") D F21 Q
F21 I SD('FM(%J,PG,LN,TP)) S:'SD(LC(PG,"FILL")) LN=LN+1
--> S:LN>62 LC(PG,"FILL")="" Q:SD(LC(PG,"FILL")) G F21
S FM(%J,PG,LN,TP)=TEXT S:LC(PG)<LN LC(PG)=LN W:'(LN#10) "." Q

```

***** INDEX OF T3CF1 *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

```

%                    HEADER+1*,HEADER+2*,HEADER+3I
%1                   HEADER+1*,HEADER+3I

```

```

%99          INIT+1*,TAB+1*,HEADER* HEADER+3!
ALPHA        INIT+1*,END!,L2
>> BP        INIT+2*,HEADER
C            INIT+1*,END!
C1           END+1!,L2*
C2           END+1!,WR1*,WR1+1
C3           END+1!,WR0*,WR1,WR1+1,L2*,F2,F2+1,F2+2
C4           END+1!,WR0*,WR1+1,L2*
CNT          END!,DATA+1*,DE+1*
D            END+2!
D1           END!
D2           END!
D3           END!
DE           END!,DE+1*
>> FN        INIT,BEG+1,HEADER,HEADER+1,HEADER+2,DATA+2,WR1,WR1+1,DE
              DE+1,F2,F2+1,F2+2
              INIT+2*,END+2!
HCT          INIT+1*,HEADER+1*
>> HD        END+2!,DATA+2*,DE,DE+1
I            END!,DE+1*
I1           END!,DATA+1*,WR0*,L2
I2           END!,WR1+1*,F2,F2+1,F2+2
I3           END!,L2*
I4           END!
I5           END+2!
I6           END+2!,WR0*
II           INIT+1*,HEADER*,HEADER+1
>> LC        BEG+1,HEADER*,HEADER+1*,HEADER+2*,F21*,F21+1*
>> LC(       INIT+1*,HEADER+2
>> LINE      INIT+1*,END+1!,F2+2*,F2+3,F21*,F21+1
LN           INIT+2*,HEADER+2
>> LP        DATA+1*,WR1*
>> LPG       INIT+1*,END+2!
N            INIT+2*,END+1!
OLN          INIT+2*,END+2!
P2           INIT+1*,END+2!,HEADER+1*,WR1*,F21,F21+1
PG           INIT+1*,END+1!
POS          END!,DE*
SEC          END!,DE*,DE+1
>> SECN      DE+1*
SHT          INIT+2*,END+2!
SP           INIT+1*,TAB+1*,TAB+2!
TAB          END+2!,HEADER+3!
>> TB        INIT+1*,INIT+2*
TB1          TAB+1,TAB+2!,HEADER+2*,HEADER+3!
TBX          TAB+2*,HEADER+2*,HEADER+3!
TCF(        WR0,WR1+1!,DE+1*,L2
TEXT         END+1!,F2*,F21+1
TLN          END+2!
TP           END+1!,F2+1*,F2+3,F21,F21+1
U            INIT+1*,END+2!,HEADER,HEADER+2,WR1
V            END+2!
X            END+2!
X1B          END+1!

```

X1E	END+1!
XB	END+1!
XE	END+1!

GLOBAL VARIABLES (* Changed ! Killed)

^FM(INIT+2!, BEG+1*, HEADER*, HEADER+1*, HEADER+2*, F21, F21+1*
^FORMS(INIT, HEADER, HEADER+1, HEADER+2, DATA+2, WR1, WR1+1, DE, DE+1, F2 F2+1, F2+2

NAKED GLOBALS

^(0)	HEADER, HEADER+2
------	------------------

LABEL REFERENCES

DE	DATA+2
F2	WR1+1
F21	F2+3, F21
L2	WR0
TAB	HEADER+2
WR1	WR0, L2

EXTERNAL REFERENCES

NONE


```

T3CF2 ;THIS PROGRAM PRINTS OUT CA FORM; AJF; 9/87 : 16 Nov 87 3:46 AM
INIT I 'SD(^FM)!(^SD(^FM($J,0))) W !,"A form must be compiled first!" q
      I 'SD(FN) W !,"A form must be selected!" q
      I ^FM($J,0)'=FN W !,"INVALID FORM!" Q
      S U="^", (TB,PG,LN)="^"
INST I SD(^FORMS(FN,2,0)) W !," Do you want to print the instructions ? <N> "
--> R ANS S:ANS="^" ANS="N" D:SE(ANS,1){ "Y"!($E(ANS,1){ "Y") ^T3CFI W #
      S POS="^",N=1,ALPHA="abcdefghijklmnopq",CONSR= DD("FUNC",116,1)
      S BOLD="DD("FUNC",114,1),COND="DD("FUNC",115,1),(P2,SHT)="^", (OLN,HT)=0
BEG D ^%IS O IO U IO D PRINT
CLOSE W # W *13 C IO
      K (DIC,DA,D0,DUZ,DTIME) Q
PRINT ;PRINTOUT FORMS
      S PG="^" F %88=1:1 S PG=SO(^FM("B",$J,PG)) Q:PG="^" S (TLN,LN,OLN)=1
--> W # X BOLD,XB W ^FM($J,PG,1,1) D PRI
      QUIT
PRI F %2=1:1 S LN=SO(^FM($J,PG,LN)) Q:LN="^" S TEXT=^FM($J,PG,LN,1) D PRI1
      Q
PRI1 I LN=1 W ! S OLN=LN
      I LN>OLN F I6=OLN+1:1:LN W !
      S OLN=LN
      I SE(^FM($J,PG,LN,1),1)="*" D P1 Q
      W SE(^FM($J,PG,LN,1),1,132) ;K ^FM($J,PG,LN,1)
      I LN>64 S OLN=0 W #
      Q
P1 S SHT=$P(^FM($J,PG,LN,1),"^",2) ;K ^FM($J,PG,LN,1)
      X XB W SHT,! X XE S OLN=OLN+1 Q

```

***** INDEX OF T3CF2 *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed : Killed)

>> %2	PRI*
>> %88	PRINT+1*
>> ALPHA	INST+1*
>> ANS	INST*
>> BOLD	INST+2*,PRINT+1
>> COND	INST+2*
>> CONSR	INST+1*
>> D0	CLOSE+1
>> DA	CLOSE+1
>> DIC	CLOSE+1
DTIME	CLOSE+1
DUZ	CLOSE+1
>> FN	INIT+2,INST
>> HT	INST+2*
>> I6	PRI1+1*
IO	BEG,CLOSE
>> LN	INIT+3*,PRINT+1*,PRI*,PRI1,PRI1+1,PRI1+2,PRI1+3,PRI1+4, PRI1+5,P1
>> N	INST+1*
>> OLN	INST+2*,PRINT+1*,PRI1*,PRI1+1,PRI1+2*,PRI1+5*,P1+1*
>> P2	INST+2*
>> PG	INIT+3*,PRINT+1*,PRI,PRI1+3,PRI1+4,P1

```

>> POS          INST+1*
>> SHT          INST+2*,P1*,P1+1
>> TB           INIT+3*
>> TEXT         PRI*
>> TLN          PRINT+1*
>> U            INIT+3*
>> XB           PRINT+1,P1+1
>> XE           P1+1

```

GLOBAL VARIABLES (* Changed ! Killed)

```

^DD(          INST+1,INST+2
^FM(          INIT+2,PRINT+1,PRI,PRI+3,PRI+4,P1
^FORMS(       INST

```

NAKED GLOBALS

NONE

LABEL REFERENCES

```

P1            PRI+3
PRI           PRINT+1
PRI1          PRI
PRINT         BEG

```

EXTERNAL REFERENCES

```

^%IS          BEG
^T3CFI        INST

```

```

T3CFB ;THIS PROGRAM CREATES A TEMPORARY GLOBAL FOR BLANK FORMS ; 25 Aug 88
S DIC="1002",DIC(0)="A",DR=".01" D "DIC Q:Y<1 S FN=Y
INIT Q:'SD(FN) I 'SD("FORMS(FN,0)) W !,!, " INVALID SELECTION " Q
S U="",(TB,PG,LN,LC)="",POS="",N=1,ALPHA="abcdefghijklmnopq"
S C=0,(HD,LINE,SP)=" F %99=1:1:132 S LINE=LINE " "
S (P2,SHT)="",(OLN,HCT)=0,LP=60,BF=1,TB=1 K "FM
BEG W !,"Compiling forms [... d HEADER,DATA,STORE W "],,1
S "FM("B",$J,1)=LC(1),"FM("B",$J,2)=LC(2),"FM($J,0)=FN D "T3CF2
END K CNT,DE,SECN,SEC,I1,I2,I3,I4,I5,ALPHA,C,D1,D2,D3
K C1,C2,C3,C4,LN,OLN,POS,TEXT,TP,X1B,X1E,XB,XE
K I6,N,P2,PG,SHT,TAB,TLN,U,V,D,I,X,HCT,I1
Q
TAB ;CREATES A TAB CHARACTER
S SP="" Q:'SD(TB1) F %99=1:1:TB1 S SP=SP_ "
S TBX=SP TBX K TB1,SP Q
STORE ;CREATE ONLY ONE STRING PER LINE
F PG=1,2 S LN="" F %1=1:1 S LN=SO("FM($J,PG,LN)) Q:LN=""
--> S (TEXT,TX,BL,TAB)=" F %1=1:1 S TAB=SO("FM($J,PG,LN,TAB)) Q:TAB=""
--> S TEXT="FM($J,PG,LN,TAB) D FILL S "FM($J,PG,LN,1)=TX
Q
FILL S TXL=SL(TX) G:TAB=1 F10
I TXL=TB1(TXL>TAB) S TX=TX_ " G F10
F %99=TXL:1:TAB-1 S TX=TX_ "
F10 S TX=TX TEXT,TXL=SL(TX),CNT=0
F11 I SL(TX)>132 G:SE(TX,SL(TX))'=" " F12 S TX=SE(TX,1,SL(TX)-1) G F11
F12 I SL(TX)>132 G:SP(TX," ",1)=TX!(CNT>10) F13 S CNT=CNT+1,
--> TX=SP(TX," ",1) " " SP(TX," ",2,999) G F12
F13 K:TAB=1 "FM($J,PG,LN,TAB) Q
HEADER F %99=1,2 S "FM($J,%99,BP,1)=SP("FORMS(FN,0),U,1) " "
--> SP("0),U,2),LC(%99)=BP,LC=BP
F %1=1,5 S PG=SS(%5:2,1:1),LC(PG)=LC I SD("FORMS(FN,%0)) S HD=0
--> F %1=1:1 S HD=SO("FORMS(FN,%0,HD)) Q:HD=""
--> S LC(PG)=LC(PG)+1,"FM($J,PG,LC(PG),1)="FORMS(FN,%0,HD,0) F %1=1,2
--> S LC(%)=LC(%)+1,"FM($J,%LC(%),1)=LINE,"FM($J,%LP+3,1)=LINE,
--> TBX=SP("FORMS(FN,0),U,1) " REVISION DATE: "
--> SP("0),U,3),TB1=132-SL(TBX)\2 D TAB
--> S "FM($J,%LP+4,1)=TBX,"FM("B",$J,%)=LC(%)+1
K TBX,TB1,TAB,%1,%99,% Q
DATA ;CREATES DATA FORMAT
S (CNT,I2)=0,LPG=1
F I=1:1 Q:'SD("FORMS(FN,6,"B",I)) D DE
WRO F I1=1:1 S I2=$N(TCF(I2)) Q:I2=1 S C3=TCF(I2),C4=I2 D WRI,L2
Q
WRI S C2=SP("FORMS(FN,3,C3,1,0),"",3),PG=SP("FORMS(FN,3,C3,0),U,5)
S:PG="" PG=LPG S LPG=PG
F I3=1:1:C2 D:SD("FORMS(FN,3,C3,1,I3,0)) F2 K TCF(C4)
Q
DE S SEC=SO("FORMS(FN,6,"B",I,""),SECN=SP("FORMS(FN,6,SEC,0),"",2)
F I1=1:1 S CNT=SO("FORMS(FN,3,"C",SECN,CNT)) Q:CNT=""
--> S:SD("FORMS(FN,3,CNT,0)) DE=SP("FORMS(FN,3,CNT,0),"",1),
--> TCF(DE)=CNT,SECN(I)=SECN
Q
L2 F I4=1:1 S C1=I2_SE(ALPHA,I4) Q:'SD(TCF(C1)) S C3=TCF(C1),C4=C1 D WRI
Q

```

```

F2      S TEXT=$P(^FORMS(FN,3,C3,1,I3,0),"",1)
        S TP=$P(^FORMS(FN,3,C3,1,I3,0),"",2)
        S LN=$P(^FORMS(FN,3,C3,1,I3,0),"",3)
        Q:LN=""(TP="") D F21 Q
F21     I $D(^FM($J,PG,LN,TP)) S:$D(LC(PG,"FILL")) LN=LN+1
—>     S:LN>62 LC(PG,"FILL")="" Q:$D(LC(PG,"FILL")) G F21
        S ^FM($J,PG,IN,TP)=TEXT S:LC(PG)<LN LC(PG)=LN W: '(LN#10) "." Q

```

***** INDEX OF T3CFB *****

LOCAL VARIABLES	LINE OCCURRENCES (* Changed ! Killed)
%	STORE+1*,HEADER+1*,HEADER+2*,HEADER+3!
%1	STORE+1*,HEADER+1*,HEADER+3!
%99	INIT+1*,TAB+1*,FILL+2*,HEADER*,HEADER+3!
ALPHA	INIT+1*,END!,L2
>> BL	STORE+1*
>> BP	INIT+2*,HEADER
C	INIT+1*,END!
C1	END+1!,L2*
C2	END+1!,WR1*,WR1+1
C3	END+1!,WR0*,WR1,WR1+1,L2*,F2,F2+1,F2+2
C4	END+1!,WR0*,WR1+1,L2*
QNT	END!,FI0*,FI2*,DATA+1*,DE+1*
D	END+2!
D1	END!
D2	END!
D3	END!
DE	END!,DE+1*
>> DIC	T3CFB+1*
>> DIC(T3CFB+1*
>> DR	T3CFB+1*
>> FN	T3CFB+1*,INIT,BEG+1,HEADER,HEADER+1,HEADER+2,DATA+2,WR1 WR1+1,DE,DI 1,F2,F2+1,F2+2
HCT	INIT+2*,EN +2!
>> HD	INIT+1*,HEADER+1*
I	END+2!,DATA+2*,DE,DE+1
I1	END!,DE+1*
I2	END!,DATA+1*,WR0*,L2
I3	END!,WR1+1*,F2,F2+1,F2+2
I4	END!,L2*
I5	END!
I6	END+2!
II	END+2!,WR0*
>> LC	INIT+1*,HEADER*,HEADER+1
>> LC(BEG+1,HEADER*,HEADER+1*,HEADER+2*,F21*,F21+1*
>> LINE	INIT+1*,HEADER+2
LN	INIT+1*,END+1!,STORE+1*,FI3,F2+2*,F2+3,F21*,F21+1
>> LP	INIT+2*,HEADER+2
>> LPG	DATA+1*,WR1*
N	INIT+1*,END+2!
OLN	INIT+2*,END+1!
P2	INIT+2*,END+2!
PG	INIT+1*,END+2!,STORE+1*,FI3,HEADER+1*,WR1*,F21,F21+1

```

POS      INIT+1*,END+1!
SEC      END!,DE*
SECN     END!,DE*,DE+1
>> SECN( DE+1*
SHT      INIT+2*,END+2!
SP       INIT+1*,TAB+1*,TAB+2!
TAB      END+2!,STORE+1*,FILL,FILL+1,FILL+2,FI3,HEADER+3!
>> TB    INIT+1*,INIT+2*
TB1      TAB+1,TAB+2!,HEADER+2*,HEADER+3!
TBX      TAB+2*,HEADER+2*,HEADER+3!
TCF(     WR0,WR1+1!,DE+1*,L2
TEXT     END+1!,STORE+1*,FI0,F2*,F21+1
TLN      END+2!
TP       END+1!,F2+1*,F2+3,F21,F21+1
>> TX    STORE+1*,FILL,FILL+1*,FILL+2*,FI0*,FI1*,FI2*
>> TXL   FILL*,FILL+1,FILL+2,FI0*
U        INIT+1*,END+2!,HEADER,HEADER+2,WR1
V        END+2!
X        END+2!
X1B      END+1!
X1E      END+1!
XB       END+1!
XE       END+1!
>> Y     T3CFB+1

```

GLOBAL VARIABLES (* Changed ! Killed)

```

^FM(      INIT+2!,BEG+1*,STORE+1*,FI3!,HEADER*,HEADER+1*,HEADER+2*
          F21,F21+1*
^FORMS(    INIT,HEADER,HEADER+1,HEADER+2,DATA+2,WR1,WR1+1,DE,DE+1,F2
          F2+1,F2+2

```

NAKED GLOBALS

```

^(0)      HEADER,HEADER+2

```

LABEL REFERENCES

```

DE        DATA+2
F2        WR1+1
F21       F2+3,F21
FI0       FILL,FILL+1
FI1       FI1
FI2       FI1,FI2
FI3       FI2
FILL      STORE+1
L2        WR0
TAB       HEADER+2
WR1       WR0,L2

```

EXTERNAL REFERENCES

```

^DIC      T3CFB+1
^T3CF2    BEG+1

```

```

T3CFCK ;DMP; IS LAST CLAIM FILED IS IN PROPER SEQUENCE? ; 09 Nov 87 2:43 PM
INIT S msg=" ",y=24,FORM="",U="" G:$D(DA) EX
S:$D(om) om="" g:om="" EX D EX
S DATE=$P(obi(5),db,2),DATE=$P(DATE,".",1),CF=om,FN=$P(^FORMS(CF,0),U,1)
S FORM=$P(^FORMS(CF,0),U,5) I FORM=""!(DATE="") S om="" g EX
;FORM NOT DEFINED OR IMPROPER DATE
CHFST ;CHECK TO SEE IF FIRST ENTRY
I '$D(^EMPLOY(NAMEPTR,"ICC","C",DATE)) S FIRST="" G FIRST
CKENTY ;CHECK TO SEE IS A CA-1, CA-2, OR CA-6 IS ALREADY ENTERED FOR THIS CASE
I '(FN="CA-1"!(FN="CA-2"!(FN="CA-6")) G LAST
S IP="" ? I=1:1 S IP=$O(^EMPLOY(NAMEPTR,"ICC","B",CF,IP)) Q:IP=""
I $D(^EMPLOY(NAMEPTR,"ICC","C",DATE,IP)) S ERROR=""
I $D(ERROR) s msg=FN " already entered for this claim.",om="",FORM=""
--> D hprt^ptrslot2 G EX
LAST S LAST="" I $D(^ICC(DA,200)) S LAST=$P(^ICC(DA,200),U,9)
G:LAST="" CKSEQ
F I=1:1 S LAST=$O(^EMPLOY(NAMEPTR,"ICC","C",DATE,LAST)) Q:LAST=""
--> S IP=LAST
S:$D(IP) LAST=$P(^EMPLOY(NAMEPTR,"ICC",IP,0),U,1) G:LAST="" CKSEQ
;
CHECK ; VERIFY NEXT FORM
FIRST S: '(FN="CA-1"!(FN="CA-2"!(FN="CA-16"!(FN="CA-6"!(FN="CA-17"))
--> msg="This is not a top level form (e.g. CA-1, CA-2, CA-16, CA-6)",
--> om="",FORM="" D hprt^ptrslot2 g EX
CKSEQ ;CHECK SEQUENCE
I '$D(^FORMS(LAST,7,"B",CF))
--> s msg="Select from the following forms:",om="",FORM="",NEXT=""
--> F %=1:1 S NEXT=$O(^FORMS(LAST,7,"B",NEXT)) Q:NEXT=""
--> S msg=msg "(" $P(^FORMS(NEXT,0),U,1) " ",NFLAG=1
I $D(NFLAG) s y=23 d hprt^ptrslot2 s y=24
EX K DATE,CF,LAST1,LAST,I,FIRST,FN,IP,NEXT,NFLAG.ERROR K:FORM="" FORM q

```

***** INDEX OF T3CFCK *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

CF	INIT+2*,CKENTRY+2,EX!
>> DA	INIT,LAST
DATE	INIT+2*,CHFST+1,CKENTRY+2,LAST+1,EX!
ERROR	CKENTRY+2*,EX!
FIRST	CHFST+1*,EX!
FN	INIT+2*,CKENTRY+1,EX!
FORM	INIT*,INIT+2*,FIRST*,EX!
I	CKENTRY+2*,LAST+1*,EX!
IP	CKENTRY+2*,LAST+1*,LAST+2,EX!
LAST	LAST*,LAST+1*,LAST+2*,EX!
LAST1	EX!
>> NAMEPTR	CHFST+1,CKENTRY+2,LAST+1,LAST+2
NEXT	EX!
NFLAG	EX!
>> U	INIT*,LAST*,LAST+2

GLOBAL VARIABLES (* Changed ! Killed)

EMPLOY(CHFST+1,CKENTRY+2, LAST+1, LAST+2
ICC(LAST

NAKED GLOBALS

NONE

LABEL REFERENCES

CKSEQ	LAST, LAST+2
EX	INIT, INIT+2
FIRST	CHFST+1
LAST	CKENTRY+1

EXTERNAL REFERENCES

hprrptrslot2	FIRST
--------------	-------

```

T3CFI ;THIS PROGRAM PRINTS OUT INSTRUCTIONS FOR CA FORMS;DMP; 9/87
BEG   Q:'$D(FN) I '$D(^FORMS(FN,0)) W !,!, " INVALID SELECTION " Q
      I '$D(^FORMS(FN,2)) W !,"There are no instructions for this form " q
      S BOLD=^DD("FUNC",114,1),(P2,SHT)="",(OLN,HCT)=0
OPEN  D ^%IS O IO U IO W # D INST
CLOSE W *13 C IO
      K CNT,HCT,I,OLN,SHT,XB,XE,BOLD,RMSDF,HTC,IOPAR,IOSL
      K IOST,IOT,IOM,IO,IOBS,IOF Q
INST  ;THIS SECTION PRINT OUT INSTRUCTION
      X BOLD,XB W ^FORMS(FN,2,1,0),! X XE
      S CNT=$P(^FORMS(FN,2,0),"^",3),OLN=1
      F I=2:1:CNT S INS=^FORMS(FN,2,1,0),LENG=$L(INS) D S1
      K SECN,SEC,P2,I5,I6,SP,EP,LENG,INS
      Q
S1    S SP=0,EP=133
      F I5=1:1 S P2=$E(INS,SP,LENG) D S4 Q:$L(P2)<133 D S2,S3
      C
S2    F I6=EP:-1 Q:$E(P2,I6)=" "
      Q
S3    I $E(P2,1)=" " S P2=$E(P2,2,999)
      W $E(P2,1,I6-1),! S SP=SP+I6,HCT=HCT+1 W:HCT=60 # Q
S4    Q:$L(P2)>80 I $E(P2,1)=" " S P2=$E(P2,2,999)
      W P2,! S HCT=HCT+1 W:HCT=60 #
      Q

```

***** INDEX OF T3CFI *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

BOLD	BEG+2*,CLOSE+1!,INST+1
CNT	CLOSE+1!,INST+2*,INST+3
EP	INST+4!,S1*,S2
>> FN	BEG,INST+1,INST+2,INST+3
HCT	BEG+2*,CLOSE+1!,S3+1*,S4+1*
HTC	CLOSE+1!
I	CLOSE+1!,INST+3*
I5	INST+4!,S1+1*
I6	INST+4!,S2*,S3+1
INS	INST+3*,INST+4!,S1+1
IO	OPEN,CLOSE,CLOSE+1!
IOBS	CLOSE+1!
IOF	CLOSE+1!
IOM	CLOSE+1!
IOPAR	CLOSE+1!
IOSL	CLOSE+1!
IOST	CLOSE+1!
IOT	CLOSE+1!
LENG	INST+3*,INST+4!,S1+1
OLN	BEG+2*,CLOSE+1!,INST+2*
P2	BEG+2*,INST+4!,S1+1*,S2,S3*,S3+1,S4*,S4+1
RMSDF	CLOSE+1!
SEC	INST+4!
SECN	INST+4!
SHT	BEG+2*,CLOSE+1!

SP	INST+4!,S1*,S1+1,S3+1*
XB	CLOSE+1!,INST+1
XE	CLOSE+1!,INST+1

GLOBAL VARIABLES (* Changed ! Killed)

^DD(BEG+2
^FORMS(BEG,INST+1,INST+2,INST+3

NAKED GLOBALS

NONE

LABEL REFERENCES

INST	OPEN
S1	INST+3
S2	S1+1
S3	S1+1
S4	S1+1

EXTERNAL REFERENCES

^%IS	OPEN
------	------

```

T3CFILE;DMP;THIS ROUTINE STORES THE CLAIM FORMS IN EMPLOY DATA FILE ; 02 Dec 87
INIT D EX G:('SD(DA)#10:('SD(DO)#10)) EX S:('SD(DO)#10 DO=DA
S CF=ptg(903,iform,1),CF=So('FORMS("B",CF,"") Q:CF=""
S U="",DA=DO,EMP=SP('ICC(DA,0),U,3),DATE=SP('ICC(DA,0),U,2)
S DATE=SP(DATE,".",1),IDN="",LV=0,ND="DA",FORM=iform
G:EMP="!"(DATE="") EX
I 'SD('EMPLOY(EMP,"ICC",0)) S CL=1 G FILE
I 'SD('EMPLOY(EMP,"ICC","C",DATE)) S CL=SP('EMPLOY(EMP,"ICC",0),U,4)+1
--> G FILE
S CL="" F %1:1 S CL=So('EMPLOY(EMP,"ICC","C",DATE,CL)) Q:CL=""
--> Q:SD('EMPLOY(EMP,"ICC","B",CF,CF))
I CL="" S CL=SP('EMPLOY(EMP,"ICC",0),U,4)+1 G FILE
G:SD(ERFLAG) FILE G EX
;
FILE S msg="Warning once a form is filed in the employee file it cannot be
----> edited!!!! Do you wish to continue ? <N> ",y=23 d hprt'ptrslot2
R ANS G:SE(ANS,1)'="Y"&('SE(ANS,1)'="Y") EX
F1 S msg="Filing " Sp('FORMS(CF,0),U,1) " [.",y=24 d hprt'ptrslot2
RSET D T3COPD ;SET UP COP DATA ELEMENTS
LOOP F %1:1 S IDN=So('ptg(903,FORM,12,IDN)) Q:IDN="" D GSET W:('%10) ".
S:LV=0 SP('EMPLOY(EMP,"ICC",CL,200),U,9)=CF,
----> 'EMPLOY(EMP,"ICC","B",CF,CL)="",SP('ICC(DO,200),U,9)=CF,
----> SP('EMPLOY(EMP,"ICC",CL,200),U,9)=CF,
----> SP('EMPLOY(EMP,"ICC",CL,200),U,10)="",SP('ICC(DO,200),U,10)=""
END G:SD(SUB)>1 MFLOS
S:SD('EMPLOY(EMP,"ICC",0)) 'EMPLOY(EMP,"ICC",0)=""1004.2IPA"
D:SD(ERFLAG) SP('EMPLOY(EMP,"ICC",0),U,3)=CL,
----> SP('EMPLOY(EMP,"ICC",0),U,4)=CL ;INCREMENTS COUNTER
F %2=8,12,101,211,212,222,250,252,306,311,314,402,502,
----> 505,506,507,508,509,510,511,512,520,521,522,523,524 K 'ICC(DO,13)
F %3=801,804,805,812,1600,1602,1603,1604,1605,1606,1607,1608,
----> 1700,1701,1702,1703,1704,1706,1707,1709,1710,1709,2006,2007,
----> 2008,2009,2010,2011,700,704 K 'ICC(DO,13)
S:SD('ICC(DO,303)) 'ICC(DO,303)=SP('ICC(DO,303),U,1,6)
S:SD('ICC(DO,515)) 'ICC(DO,515)=SP('ICC(DO,515),U,1,4)
W "!" D EX S FILED="",DA=DO,data="" K ERFLAG Q
EX K ND,IP,LAST,LASTI,FILED,NODE,EMP,YGLO,XGLO,IDN,NAME,CL
K %2,%1,%3,SUB,LVX,%Y,%X1,LV,DATE,I,% Q
Q ;
GSET ;FILE EMPLOYEE (YGLO) FROM ICC MODULE (XGLO)
Q:SD('ptg(903,FORM,12,IDN,28))
S NODE=ptg(903,FORM,12,IDN,28),XGLO=NODE D NAME Q:NAME=""
I NODE="nowhere"!(NODE="")
----> S NODE=SP('ptg(903,FORM,12,IDN,16),U,11),XGLO="" G:NODE?3N.E GSUB Q
S NODE=SP(SP(NODE,"",2),".",1),NODE="" NODE Q:SD(NODE)
S YGLO=SP('EMPLOY(EMP,"ICC",CL) SP(XGLO,NODE),%X="S %X1=" XGLO
S %Y="S " YGLO "-%X1" X %X,%Y Q
GSUB S N1=ptg(903,FORM,12,IDN,99),SUB(NODE)=IDN_U+SP('DO(1004.2,N1,0),U,2)
K N1 Q
Q ;
MFLOS S LV=LV+1,FORM="" I LV>1 F %2=LV:-1:1 S DA(1)=DA(12-1)
S DA(1)=DA,DA=1,ND="DA(" LV ")"
F %1:1 S FORM=So(SUB(FORM)) Q:FORM="" S IDN="" F %1:1:1
----> S IDN=So('ptg(903,FORM,12,IDN)) Q:IDN="" D MF!

```



```
>> U          INIT+4,INIT+6,END+4
>> XGLO       GSET+1*,GSET+3,MFL+1*
>> YGLO       GSET+3*
```

GLOBAL VARIABLES (* Changed ! Killed)

```
^EMPLOY(      INIT+3,INIT+4,INIT+5,INIT+6,LOOP+1*,END+1*
^ICC(         LOOP+1*,END+2!,END+3!,END+4*
```

NAKED GLOBALS

NONE

LABEL REFERENCES

```
END          MFLDS+4
EX           INIT,INIT+2,INIT+7,FILE+1,END+5
FILE         INIT+3,INIT+4,INIT+6,INIT+7
GSUB         GSET+1
MFLSUB       MFL+1
MFLDS        END
NAME         GSET+1
```

EXTERNAL REFERENCES

```
^XTCORP      MFL+4
^TCCPD       RSET
```

```

T3CFSEL;DMP;Selected LOOK UP FOR FORMS ; 12 Nov 87  4:09 AM
S DIC=1004,U="",DIE="EMPLOY(",DUZ(0)="01",DUZ=0,DTIME=600
S (DR,DR(1,1004))=".01;2000;";DR(2,1004.2)=".01;";
S DIC(0)="AQELM",ICC="ICC" D T2GUSER
D DIC Q:'SD(X):(Y<0) S E0=+Y,DA=E0 D DIE
Q:'SD(D1)!(SD(D)) S E1=D1,FN=+D Q:FN=""
S GLO="EMPLOY("_E0_",ICC,"_E1_",

```

***** INDEX OF T3CFSEL *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed : Killed)

```

>> D                    T3CFSEL+2
>> D1                   T3CFSEL+2
>> DA                   T3CFSEL+2*
>> DIC                  T3CFSEL+1*
>> DIC(                 T3CFSEL+1*
>> DIE                  T3CFSEL+1*
>> DR                   T3CFSEL+1*
>> DR(                  T3CFSEL+1*
DTIME                  T3CFSEL+1*
DUZ                    T3CFSEL+1*
>> DUZ(                 T3CFSEL+1*
>> E0                   T3CFSEL+2*
>> E1                   T3CFSEL+2*
>> FN                   T3CFSEL+2*
>> GLO                  T3CFSEL+2*
>> ICC                  T3CFSEL+1*
>> U                    T3CFSEL+1*
>> X                    T3CFSEL+2
>> Y                    T3CFSEL+2

```

GLOBAL VARIABLES (* Changed : Killed)

NONE

NAKED GLOBALS

NONE

LABEL REFERENCES

NONE

EXTERNAL REFERENCES

```

DIC                    T3CFSEL+2
DIE                    T3CFSEL+2
T2GUSER                T3CFSEL+1

```

```

T3COMP ;DMP; STORES COMP INFORMATION IN ^INJ(1009 ; 03 Nov 87 11:39 AM
INIT NEW PC,FC,EC,INC,DATA,DATE4,COMP D KILL S U=""
I ^SD(DT) S X=$ZD(+SH),%DT="" D ^%DT S DT=Y
S COMP=$S($D(D0):D0,$D(DA):DA,1:"") G EX:COMP="",EX:^SD(^ICC(COMP,220))
S:^SD(^INJ(1009,0)) ^INJ(1009,0)="COP DATA^1009P^^"
S DATA=^ICC(COMP,251),DATE4=$P(^ICC(COMP,251),U,1),DATE4=$P(DATE4,".",1)
G:DATE4="" EX
COMP ;SET COMP INFORMATION
S FC=$S($P(^INJ(1009,0),U,3)<1:1,1:$P(^INJ(1009,0),U,3)+1)
S PC=$S($D(^INJ(1009,"B",COMP)):$O(^INJ(1009,"B",COMP,"")),1:FC)
G:$D(^INJ(1009,PC)) CM2
S ^INJ(1009,PC,0)=COMP U $P(^ICC(COMP,0),U,2,3)
S ^INJ(1009,"B",COMP,PC)="^"
S ^INJ(1009,PC,1,0)="^1009.01D^1^1",INC=1,EC=1,
S ^INJ(1009,PC,1,INC,0)=DATA,^INJ(1009,"C",DATE4,PC,INC)="^"
S $P(^INJ(1009,0),U,3,4)=FC,$P(^INJ(1009,0),U,4)=FC G EX
CM2 S:^SD(^INJ(1009,PC,1,0)) ^INJ(1009,PC,1,0)="1009.01D^^"
S EC=$P(^INJ(1009,PC,1,0),U,3)+1,INC=$S($D(^INJ(1009,"C",DATE4,PC))):
--> $O(^INJ(1009,"C",DATE4,PC,"")),1:EC)
S ^INJ(1009,PC,1,INC,0)=DATA,^INJ(1009,"C",DATE4,PC,INC)="^",
S ^INJ(1009,"B",COMP,PC)="^" S:INC=EC $P(^INJ(1009,PC,1,0),U,3)=EC,
S $P(^INJ(1009,PC,1,0),U,4)=$P(^INJ(1009,PC,1,0),U,4)+1
EX D KILL Q ;
KILL K PC,FC,EC,INC,DATA,DATE4,COMP Q

```

***** INDEX OF T3COMP *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

```

>> %DT                INIT*
COMP                INIT,INIT+1*,INIT+3,COMP+2,CM1,CM2+1,KILL!
>> D0                INIT+1
>> DA                INIT+1
DATA                INIT,INIT+3*,CM1,CM2+1,KILL!
DATE4                INIT,INIT+3*,CM1,CM2,CM2+1,KILL!
DT                INIT*
EC                INIT,CM1*,CM2*,CM2+1,KILL!
FC                INIT,COMP+1*,COMP+2,CM1,KILL!
INC                INIT,CM1*,CM2*,CM2+1,KILL!
PC                INIT,COMP+2*,COMP+3,CM1,CM2,CM2+1,KILL!
>> U                INIT*,INIT+3,COMP+1,CM1,CM2,CM2+1
>> X                INIT*
>> Y                INIT

```

GLOBAL VARIABLES (* Changed ! Killed)

```

^ICC(                INIT+1,INIT+3,CM1
^INJ(                INIT+2*,COMP+1,COMP+2,COMP+3,CM1*,CM2*,CM2+1*

```

NAKED GLOBALS

NONE

LABEL REFERENCES

CM2 COMP+3
EX INIT+1, INIT+3, CM1
KILL INIT, EX

EXTERNAL REFERENCES

^3DT INIT

```

T3COP ;DMP* STORES COMP INFORMATION IN INJ(1008 ; 03 Nov 87 11:26 AM
INIT NEW PC,FC,EC,INC,COP,DATA,DATE3 D KILL S U="" I '$D(DT)
S X=$ZD(+SH),%DT="" D %DT S DT=Y
S COP=$S($D(D0):D0,$D(DA):DA,1:"") G:COP="" EX G:'$D(^ICC(COP,220)) EX
S:'$D(^INJ(1008,0)) ^INJ(1008,0)="COP DATA^1008P^^"
S DATA=^ICC(COP,220),DATE3=$P(^ICC(COP,220),U,1),DATE3=$P(DATE3,".",1)
G:DATE3="" EX
COPS ;SET COP INFORMATION
S FC=$S($P(^INJ(1008,0),U,3)<1:1,1:$P(^INJ(1008,0),U,3)+1)
S PC=$S($D(^INJ(1008,"B",COP)):SC(^INJ(1008,"B",COP,"")),1:FC)
G:$D(^INJ(1008,PC)) CP2
CP1 S ^INJ(1008,PC,0)=COP U $P(^ICC(COP,0),U,2,3)
S ^INJ(1008,"B",COP,PC)="",^INJ(1008,PC,1,0)="1008.01D^1^1"
S INC=1,EC=1,^INJ(1008,PC,1,INC,0)=DATA,^INJ(1008,"C",DATE3,PC,INC)="
S:P(^INJ(1008,0),U,3,4)=FC,$P(^INJ(1008,0),U,3)=FC G EX
CP2 S:'$D(^INJ(1008,PC,1,0)) ^INJ(1008,PC,1,0)="1008.01D^^"
S EC=$P(^INJ(1008,PC,1,0),U,3)+1
S INC=$S($D(^INJ(1008,"C",DATE3,PC))):
----> $O(^INJ(1008,"C",DATE3,PC,"")),1:EC)
S ^INJ(1008,PC,1,INC,0)=DATA,^INJ(1008,"C",DATE3,PC,INC)="
S ^INJ(1008,"E",COP,PC)=" S:INC=EC $P(^INJ(1008,PC,1,0),U,3)=EC,
----> $P(^INJ(1008,PC,1,0),U,4)-$P(^INJ(1008,PC,1,0),U,4)+1
EX D KILL Q ;-----;
KILL K PC,FC,EC,INC,DATA,DATE3,COP Q

```

***** INDEX OF T3COP *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

```

>> DT                INIT*
>> COP               INIT,INIT+1*,INIT+3,COPS+2,CP1,CP2+1,KILL!
>> D0                INIT+1
>> DA                INIT+1
>> DATA             INIT,INIT+3*,CP1,CP2+1,KILL!
>> DATE3             INIT,INIT+3*,CP1,CP2,CP2+1,KILL!
>> DT                INIT*
>> EC                INIT,CP1*,CP2*,CP2+1,KILL!
>> FC                INIT,COPS+1*,COPS+2,CP1,KILL!
>> INC               INIT,CP1*,CP2*,CP2+1,KILL!
>> PC                INIT,COPS+2*,COPS+3,CP1,CP2,CP2+1,KILL!
>> U                 INIT*,INIT+3,COPS+1,CP1,CP2,CP2+1
>> X                 INIT*
>> Y                 INIT

```

GLOBAL VARIABLES (* Changed ! Killed)

```

^ICC(                INIT+1,INIT+3,CP1
^INJ(                INIT+2*,COPS+1,COPS+2,COPS+3,CP1*,CP2*,CP2+1*

```

NAKED GLOBALS

NONE

LABEL REFERENCES

CP2 COPS+3
EX INIT+1, INIT+3, CP1
KILL INIT, EX

EXTERNAL REFERENCES

^%DT INIT

```

T3COPD ;DMP;CREATES COP NOT TO EXCEED DATE1 ; 15 Nov 87 11:19 PM
INIT N I D EX S (H,H1,H2,H3,PT)="",U="" G:'SD(D0) EX
COP I 'SD(^ICC(D0,220))
S DATE1=$P(^ICC(D0,0),U,2),DATE1=$P(DATE1,".",1)
S $P(^ICC(D0,220),U,1)=DATE1
S DATE1=$P(^ICC(D0,220),U,1),DATE1=$P(DATE1,".",1)
D:$P(^ICC(D0,220),U,2)=" NTEX
S:$D(^ICC(D0,221))&($P(^ICC(D0,220),U,4)="")
S $P(^ICC(D0,220),U,4)=$P(^ICC(D0,221),U,3) ;COP RETURN
S:$D(^ICC(D0,1700)) H=$P(^ICC(D0,1700),U,3),H1=$P(^ICC(D0,1700),U,16)
----> PT=$P(^ICC(D0,1700),U,7)
S:$D(^ICC(D0,300)) H2=$P(^ICC(D0,300),U,6),H3=$P(^ICC(D0,300),U,7)
S $P(^ICC(D0,220),U,3)=$S(H'="":H,H2'="":H2,1:8)
----> , $P(^ICC(D0,300),U,6)=$P(^ICC(D0,220),U,3) ;RESET REGULAR HOURS
I PT="!"(PT("F") S $P(^ICC(D0,220),U,5)=$P(^ICC(D0,220),U,3)
----> , $P(^ICC(D0,300),U,7)=" " ;FULL TIME
I PT("P" S $P(^ICC(D0,220),U,5)=$S(H1'="":H1,H'="":H,H3'="":H3,1:"")
----> , $P(^ICC(D0,300),U,7)=$P(^ICC(D0,220),U,5) ;PARTTIME
EX K Y,D,M,I,MXD,MXDAYS,H,H1,H2,H3,PT,DATE1 Q
;
NTEX S Y=$E(DATE1,2,3),M=$E(DATE1,4,5),D=$E(DATE1,6,7) D YR
ADD ;ADDS 45 DAYS TO DATE1
F I=1:1:45 S MXDAYS=$P(MXD,";",M+1) S D=D+1 S:D>MXDAYS M=M+1,D=1
----> I M>13 S M=1,Y=Y+1 D YR
S:$L(M)<2 M="0" M S:$L(D)<2 D="0" D S $P(^ICC(D0,220),U,2)="2" _Y_M_D Q
YR ;SETS UP THE MAXIMUM DAYS FOR A GIVEN
I Y>87,(Y#4) S MXD=$T(DATA+2) Q
S MXD=$T(DATA+1) Q
DATA ;JAN;FEB;MAR;APR;MAY;JUN;JUL;AUG;SEP;OCT;NOV;DEC
;31;28;31;30;31;30;31;31;30;31;30;31
;31;29;31;30;31;30;31;31;30;31;30;31

```

***** INDEX OF T3COPD *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

D	EX!,NTEX*,ADD+1*,ADD+2*
>> D0	INIT,COP,COP+1,COP+2,COP+3,COP+4,COP+5,COP+6,ADD+2
DATE1	COP*,COP+1*,EX!,NTEX
H	INIT*,COP+3*,COP+4,COP+6,EX!
H1	INIT*,COP+3*,COP+6,EX!
H2	INIT*,COP+3*,COP+4,EX!
H3	INIT*,COP+3*,COP+6,EX!
I	INIT,EX!,ADD+1*
M	EX!,NTEX*,ADD+1*,ADD+2*
MXD	EX!,ADD+1,YR+1*,YR+2*
MXDAYS	EX!,ADD+1*
PT	INIT*,COP+3*,COP+5,COP+6,EX!
>> U	INIT*,COP,COP+1,COP+2,COP+3,COP+4,COP+5,COP+6
Y	EX!,NTEX*,ADD+1*,ADD+2,YR+1

GLOBAL VARIABLES (* Changed ! Killed)

^ICC(COP*,COP+1,COP+2*,COP+3,COP+4*,COP+5*,COP+6*,ADD+2*

NAKED GLOBALS

NONE

LABEL REFERENCES

DATA	YR+1, YR+2
EX	INIT
NTEX	COP+1
YR	NTEX, ADD+1

EXTERNAL REFERENCES

NONE

```

T3CPAK ;DMP;CREATE AN ICC PACKAGE ; 05 Feb 88 3:01 PM
INIT W !,22,"Creating ^ICCTMP Global. This should take approx 15 minutes.",!!
D EMP T3CPAK
DEL I $D(^ROUTINE(^T3CINIT))
--> W !,"DELETE THE <T3CIN*> ROUTINES FIRST BEFORE I CONTINUE.....",!
--> D ^RDELETE G DEL
DIF S DUZ(0)="@",X="T3C" ;CREATING PACKAGE
W !,"I am going to create a routine called '",X,"INIT'."
K ^UTILITY($J),F,DTL,Q,E
S U="^",%0,%B="",DL=0,DTL=X,DN=X_$E("INI",1.5-$L(X))
S ^UTILITY($J)=DN,X=X_"INIT"
D R1+3^DIFROM
RO W !,"1. D ^%RO TO STORE ROUTINES: T3C* AND C*, VMS FILE --<T3C.ROU>"
GO W !,"2. D ^%GO TO STORE GLOBALS:
--> ^ICCTMP, ^FORMS, ^ptg, and ^ptc IN <T3C.GLO>" ;D ^%GO
W !,"3. THEN TRANSFER THE T3C.ROU AND T3C.GLO FILES TO TAPE",! Q
;
EMP K ^ICCTMP S ^ICCTMP(1004,0)="FIELD^^2000^45"
S (^ICCTMP(1004,0,"PT",1008,.03),^ICCTMP(1004,0,"PT",1009,.03))=""
F I=.01,1,2 S ^ICCTMP(1004,0,"PT",1005,I)=""
S ^ICCTMP(1004,2000,0)="INJURY COMP AND CLAIMS 1004.2IPA^^ICC;0",
S ^ICCTMP(1004,"B","INJURY COMP AND CLAIMS",2000)=""
S (^ICCTMP(1004,"GL","ICC",0,2000),^ICCTMP(1004,"SB",1004.2,2000))=""
S X=1004.199 F I=1:1 S X=$O(^DD(X)) Q:X>1004.999
--> S %X="^DD(^_X_",",%Y="^ICCTMP(^_X_", " D %XY^RCR
Q ;

```

***** INDEX OF T3CPAK *****

```

LOCAL          LINE OCCURRENCES ( * Changed ! Killed)
VARIABLES
>> %           DIF+2*
>> %B          DIF+2*
>> %X          EMP+2*
>> %Y          EMP+2*
>> DL          DIF+2*
>> DN          DIF+2*
>> DTL         DIF+2!*
>> DUZ(        DIF*
>> E           DIF+2!
>> F           DIF+2!
>> I           EMP*,EMP+2*
>> Q           DIF+2!
>> U           DIF+2*
>> X           DIF*,DIF+1,DIF+2*,EMP+2*

```

GLOBAL VARIABLES (* Changed ! Killed)

```

^DD(           EMP+2
^ICCTMP(       EMP!*,EMP+1*
^ROUTINE(      DEL
^UTILITY(      DIF+2!*

```

NAKED GLOBALS

NONE

LABEL REFERENCES

DEL DEL

EXTERNAL REFERENCES

%XY%RCP	EMP+2
%RDELETE	DEL
R1+3%DIFROM	DIF+3
EMP%T3CPAK	INIT

T3CPYNF;DMP; STORES PAY INFORMATION IN PAY INFO FILE ; 12 Aug 88 12:12 PM

```

INIT  D KILL S U=" " I 'SD(DT) S X=$ZD(+SH),%DT=" " D %DT S DT=Y
      S PAY=$S($D(D0):D0,$D(DA):DA,1:"") Q
INC   S: 'SD(^ICC(PAY,5,0)) ^ICC(PAY,5,0)=""1005.381^0"
      S N=$P(^ICC(PAY,5,0),U,3),N=N+1,$P(^ICC(PAY,5,0),U,3,4)=N_U_N Q
PAY   S FROM=$P(FROM,"",1) Q:FROM=""
      Q:$D(^ICC(PAY,5,"B",TYPE))&($D(^ICC(PAY,5,"C",FROM))) D INC
      S ^ICC(PAY,5,N,0)=TYPE U FROM U THRU U GROSS U R U OTHER
      S ^ICC(PAY,5,"B",TYPE,N)="",^ICC(PAY,5,"C",FROM,N)="" Q
c600  ;ENTRY/EDIT LOG
      D INIT G EX: 'SD(PAY),EX:PAY="" S:$D(^ICC(PAY,3))
-->   $P(^ICC(PAY,222),U,1)=$P(^ICC(PAY,3),U,8),
-->   $P(^ICC(PAY,222),U,2)=$P(^ICC(PAY,3),U,7) d T3COPD G EX
CA1   ;STORES PAY INFORMATION
      D INIT G EX: 'SD(PAY),EX:PAY="" S (R,R1,B,B1)=""
      S:$D(^ICC(PAY,3)) B=$P(^ICC(PAY,3),U,7),R=$P(^ICC(PAY,3),U,8)
      S:$D(^ICC(PAY,222)) R1=$P(^ICC(PAY,222),U,1),B1=$P(^ICC(PAY,222),U,2)
      G:R1=R EX S:R="" $P(^ICC(PAY,3),U,7,8)=B1_U_R1
      S:R1="" $P(^ICC(PAY,222),U,1,2)=R_U_B G EX
CA2A  D INIT G EX: 'SD(PAY),EX:PAY=""
      I SD(^ICC(PAY,232)) S R="1,3,5,7,9",R1="2,3,4,5,6" F I=1:1:5
-->   S B=$P(R,"",1),B1=$P(R1,"",1) S:$P(^ICC(PAY,232),U,B1)=""
-->   $P(^ICC(PAY,231),U,B)=$P(^ICC(PAY,232),U,B1)
      S TYPE=2
      S R=$S($D(^ICC(PAY,214)):$P(^ICC(PAY,214),U,1,2),
-->   $D(^ICC(PAY,213)):$P(^ICC(PAY,213),U,1,2),$D(^ICC(PAY,231)):$P(^ICC(PAY,231),U,1,2),
-->   $D(^ICC(PAY,222)):$P(^ICC(PAY,222),U,1,2),1:"")
      I SD(^ICC(PAY,250)) S FROM=$P(^ICC(PAY,250),U,1),
-->   THRU=$P(^ICC(PAY,250),U,2), GROSS=$P(^ICC(PAY,250),U,16),OTHER=""
      G EX: 'SD(FROM),EX:FROM="" D PAYINFO G EX
CA3   D INIT G EX: 'SD(PAY),EX:PAY=""
      S R=$S($D(^ICC(PAY,231)):$P(^ICC(PAY,231),U,1,2),
-->   $D(^ICC(PAY,222)):$P(^ICC(PAY,222),U,1,2),
-->   $D(^ICC(PAY,3)):$P(^ICC(PAY,3),U,8)_U_$P(^ICC(PAY,3),U,7),1:"")
      G: 'SD(^ICC(PAY,250)) CA31
      K RO I $P(^ICC(PAY,250),U,1)=""
-->   S TYPE=2, FROM=$P(^ICC(PAY,250),U,1),
-->   THRU=$P(^ICC(PAY,250),U,2), GROSS=$P(^ICC(PAY,250),U,16), OTHER=""
      S:$P(^ICC(PAY,250),U,17)="" RO=R, R=$P(^ICC(PAY,250),U,17,18)
      D PAYINFO S:$D(RO) R=RO K RO
-->   I $P(^ICC(PAY,250),U,3)="" S TYPE=5, FROM=$P(^ICC(PAY,250),U,3),
-->   THRU=$P(^ICC(PAY,250),U,4), GROSS="", OTHER="" D PAYINFO
-->   I $P(^ICC(PAY,250),U,5)="" S TYPE=4, FROM=$P(^ICC(PAY,250),U,5),
-->   THRU=$P(^ICC(PAY,250),U,6), GROSS="", OTHER="" D PAYINFO
-->   I $P(^ICC(PAY,250),U,7)="" S TYPE=6, FROM=$P(^ICC(PAY,250),U,7),
-->   THRU=$P(^ICC(PAY,250),U,8), GROSS="", OTHER=$P(^ICC(PAY,250),U,9)
-->   D PAYINFO
CA31  I SD(^ICC(PAY,251)) I $P(^ICC(PAY,251),U,1)="" S TYPE=3,
-->   FROM=$P(^ICC(PAY,251),U,1), THRU=$P(^ICC(PAY,251),U,2),
-->   GROSS=$P(^ICC(PAY,251),U,3), OTHER="" D PAYINFO
-->   I SD(^ICC(PAY,252)) I $P(^ICC(PAY,252),U,1)=""
-->   S TYPE=7, FROM=$P(^ICC(PAY,252),U,1), THRU=$P(^ICC(PAY,252),U,2),
-->   GROSS=$P(^ICC(PAY,252),U,3), OTHER=$P(^ICC(PAY,252),U,5),
-->   R=$P(^ICC(PAY,252),U,4,5) D PAYINFO G EX ;MISC

```

```

      G EX ; _____;
CA6   G CA3 ;
CA7   G CA3 ;
CA8   G CA3 ;
EX     D KILL Q ;EXIT ROUTINE
KILL   K I,TYPE,R,R1,B,B1,FROM,THRU,RATE,BASIS,GROSS,OTHER,PAY Q

```

***** INDEX OF T3CPINF *****

LOCAL VARIABLES LINE OCCURRENCES (* Changed ! Killed)

```

>> %DT            INIT*
B                CA1+1*,CA1+2,CA2A+1*,KILL!
B1               CA1+1*,CA1+2,CA2A+1*,KILL!
BASIS            KILL!
>> D0            INIT+1
>> DA            INIT+1
DT               INIT*
FROM             PAYINFO*,CA2A+3*,CA2A+4,CA3+2*,CA3+3*,CA3+4*,CA3+5*,CA31*
                 CA31+1*,KILL!
GROSS            PAYINFO,CA2A+3*,CA3+2*,CA3+3*,CA3+4*,CA3+5*,CA31*,CA31+1*
                 KILL!
I                CA2A+1*,KILL!
>> N            INC*,PAYINFO
OTHER            PAYINFO,CA2A+3*,CA3+2*,CA3+3*,CA3+4*,CA3+5*,CA31*,CA31+1*
                 KILL!
PAY              INIT+1*,INC,PAYINFO,CA1+1,CA1+2,CA2A,CA2A+1,CA2A+2,CA2A+3
                 CA3,CA3+1,CA3+2,CA3+3,CA3+4,CA3+5,CA31,CA31+1,KILL!
R                PAYINFO,CA1+1*,CA1+2,CA2A+1*,CA2A+2,CA3+1*,CA3+2*,CA31+1*
                 KILL!
R0               CA3+2!*
R1               CA1+1*,CA1+2,CA2A+1*,KILL!
RATE             KILL!
THRU             PAYINFO,CA2A+3*,CA3+2*,CA3+3*,CA3+4*,CA3+5*,CA31*,CA31+1*
                 KILL!
TYPE             PAYINFO,CA2A+2*,CA3+2*,CA3+3*,CA3+4*,CA3+5*,CA31*,CA31+1*
                 KILL!
>> U            INIT*,INC,PAYINFO,CA1+1,CA1+2,CA2A+1,CA2A+2,CA2A+3,CA3+1
                 CA3+2,CA3+3,CA3+4,CA3+5,CA31,CA31+1
>> X            INIT*
>> Y            INIT

```

GLOBAL VARIABLES (* Changed ! Killed)

```

^ICC(            INC*,PAYINFO*,c600+1*,CA1+1,CA1+2*,CA2A+1*,CA2A+2,CA2A+3
                 CA3+1,CA3+2,CA3+3,CA3+4,CA3+5,CA31,CA31+1

```

NAKED GLOBALS

```

^ (213)          CA2A+2
^ (214)          CA2A+2
^ (222)          CA2A+2,CA3+1
^ (231)          CA2A+2,CA3+1

```

LABEL REFERENCES

CA3	CA6, CA7, CA8
CA31	CA3+1
EX	c600+1, CA1+1, CA1+2, CA2A, CA2A+4, CA3, CA31+1, CA31+2
INC	PAYINFO
INIT	c600+1, CA1+1, CA2A, CA3
KILL	INIT, EX
PAYINFO	CA2A+4, CA3+2, CA3+3, CA3+4, CA3+5, CA31, CA31+1

EXTERNAL REFERENCES

3DT

INIT

***** CROSS REFERENCE OF ALL ROUTINES *****

LOCAL VARIABLES	ROUTINES (>> not killed explicitly) (* Changed ! Killed)
%	T3CF*!, T3CF1*!, T3CFB*!, T3CFILE*, T3CPAK*
%1	T3CF*, T3CF1*!, T3CFB*!
%2	T3CF*!, T3CF2*, T3CFILE*
%3	T3CF*!, T3CFILE*
%4	T3CF*!
>> %88	T3CF2*
%98	T3CF*!
%99	T3CF*!, T3CF1*!, T3CFB*!
>> %B	T3CPAK*
>> %DT	T3COMP*, T3COP*, T3CPYNF*
%X	T3CFILE*!, T3CPAK*
%Y	T3CFILE*!, T3CPAK*
%Y1	T3CFILE*!
A1	T3CF*!
A2	T3CF*!
ALPHA	T3CF1*!, T3CF2*, T3CFB*!
>> ANS	T3CF2*, T3CFILE*
B	T3CPYNF*!
B1	T3CPYNF*!
BASIS	T3CPYNF!
BEG	T3CF*!
>> BL	T3CF*, T3CFB*
BOLD	T3CF2*, T3CF1*!
>> BP	T3CF1*, T3CFB*
C	T3CF1*!, T3CFB*!
C1	T3CF1*!, T3CFB*!
C2	T3CF1*!, T3CFB*!
C3	T3CF1*!, T3CFB*!
C4	T3CF1*!, T3CFB*!
CF	T3CFCK*!, T3CFILE*
>> CL	T3CFILE*
CNT	T3CF*!, T3CF1*!, T3CFB*!, T3CFI*!
COMP	T3COMP*!
>> COND	T3CF2*
>> CONSR	T3CF2*
COP	T3COP*!
CU	T3CF*!
D	T3CF1!, T3CFB!, T3CFSEL, T3COPD*!
>> D0	T3CF2, T3CFILE*, T3COMP, T3COP, T3COPD, T3CPYNF
D1	T3CF1!, T3CFB!, T3CFSEL
D2	T3CF1!, T3CFB!
D3	T3CF1!, T3CFB!
>> DA	T3CF2, T3CFCK, T3CFILE*, T3CFSEL*, T3COMP, T3COP, T3CPYNF
>> DA(T3CFILE*
DATA	T3COMP*!, T3COP*!
DATE	T3CFCK*!, T3CFILE
DATE1	T3COPD*!
DATE3	T3COP*!
DATE4	T3COMP*!

DE	T3CF1*!, T3CFB*!
DF1	T3CF*!
>> DIC	T3CF2, T3CFB*, T3CFSEL*
>> DIC(T3CFB*, T3CFSEL*
>> DIE	T3CFSEL*
>> DL	T3CPAK*
>> DN	T3CPAK*
>> DR	T3CFB*, T3CFSEL*
>> DR(T3CFSEL*
DT	T3COMP*, T3COP*, T3CPYNF*
DTIME	T3CF2, T3CFSEL*
DTL	T3CPAK*!
DUZ	T3CF2, T3CFSEL*
>> DUZ(T3CFSEL*, T3CPAK*
E	T3CPAK!
>> E0	T3CF, T3CFSEL*
>> E1	T3CFSEL*
EC	T3COMP*!, T3COP*!
>> ELEM	T3CF*
>> EMP	T3CFILE
END	T3CF*!
EP	T3CF1*!
ERFLAG	T3CFILE!
ERROR	T3CFCK*!
F	T3CPAK!
FC	T3COMP*!, T3COP*!
>> FILED	T3CFILE*
FIRST	T3CFCK*!
FN	T3CF*, T3CF1, T3CF2, T3CFB*, T3CFCK*!, T3CFI, T3CFSEL*
FORM	T3CFCK*!, T3CFILE*!
FROM	T3CPYNF*!
GLB	T3CF*!
>> GLO	T3CF, T3CFSEL*
GLO1	T3CF*!
GLO2	T3CF*!
GROSS	T3CPYNF*!
H	T3COPD*!
H1	T3COPD*!
H2	T3COPD*!
H3	T3COPD*!
HCT	T3CF1*!, T3CFB*!, T3CFI*!
>> HD	T3CF1*, T3CFB*
>> HT	T3CF2*
HTC	T3CFI!
I	T3CF1*!, T3CFB*!, T3CFCK*!, T3CFI*!, T3CFILE*, T3COPD*!, T3CPAK*
	T3CPYNF*!
I1	T3CF1*!, T3CFB*!
I2	T3CF1*!, T3CFB*!
I3	T3CF1*!, T3CFB*!
I4	T3CF1*!, T3CFB*!
I5	T3CF1!, T3CFB!, T3CFI*!
I6	T3CF1!, T3CF2*, T3CFB!, T3CFI*!
>> ICC	T3CFSEL*
IDN	T3CFILE*!

II	T3CF1*!,T3CFB*!
INC	T3COMP*!,T3COP*!
INS	T3CFI*!
INUM	T3CFILE*!
>> INUM(T3CFILE*
IO	T3CF2,T3CFI!
IOBS	T3CFI!
IOF	T3CF*,T3CFI!
IOM	T3CFI!
IOPAR	T3CFI!
IOSL	T3CFI!
IOST	T3CFI!
IOT	T3CFI!
IP	T3CFCK*!
LAST	T3CFCK*!
LAST1	T3CFCK!
>> LC	T3CF1*,T3CFB*
>> LC(T3CF1*,T3CFB*
LENG	T3CFI*!
>> LINE	T3CF1*,T3CFB*
LN	T3CF*!,T3CF1*!,T3CF2*,T3CFB*!
LN1	T3CF*!
LN2	T3CF*!
>> LP	T3CF1*,T3CFB*
>> LPG	T3CF1*,T3CFB*
>> LV	T3CFILE*
M	T3COPD*!
MM	T3CF*!
MULT	T3CF*!
MOD	T3COPD*!
MODAYS	T3COPD*!
N	T3CF1*!,T3CF2*,T3CFB*!,T3CPYNF*
N1	T3CFILE*!
NA	T3CF*!
>> NAME	T3CF*
>> NAMEPTR	T3CFCK
>> ND	T3CFILE*
NEXT	T3CFCK!
NFLAG	T3CFCK!
NOD	T3CF*!
>> NODE	T3CF*,T3CFILE*
OLN	T3CF1*!,T3CF2*,T3CFB*!,T3CFI*!
OTHER	T3CPYNF*!
P2	T3CF1*!,T3CF2*,T3CFB*!,T3CFI*!
PAY	T3CPYNF*!
PC	T3COMP*!,T3COP*!
PEC	T3CF*!
PG	T3CF*!,T3CF1*!,T3CF2*,T3CFB*!
POS	T3CF1*!,T3CF2*,T3CFB*!
PT	T3COPD*!
Q	T3CPAK!
R	T3CPYNF*!
R0	T3CPYNF*!
R1	T3CPYNF*!

RATE	T3CPYNF!
RMSDF	T3CFI!
SEC	T3CF1*!, T3CFB*!, T3CFI!
SECN	T3CF1*!, T3CFB*!, T3CFI!
>> SECN(T3CF1*, T3CFB*
SEL	T3CF*!
SHT	T3CF1*!, T3CF2*, T3CFB*!, T3CFI*!
SP	T3CF1*!, T3CFB*!, T3CFI*!
ST	T3CF*!
SUB	T3CFILE*!
>> SUB(T3CFILE*
SUB1	T3CFILE*!
>> SUB1(T3CFILE*
TAB	T3CF*, T3CF1!, T3CFB*!
TB	T3CF*!, T3CF1*, T3CF2*, T3CFB*
TB1	T3CF1*!, T3CFB*!
TBX	T3CF1*!, T3CFB*!
TCF(T3CF1*!, T3CFB*!
TCN	T3CF*!
TEXT	T3CF*!, T3CF1*!, T3CF2*, T3CFB*!
>> TEXT(T3CF*
THRU	T3CPYNF*!
TLN	T3CF1!, T3CF2*, T3CFB!
TP	T3CF1*!, T3CFB*!
TX	T3CF*!, T3CFB*
TXL	T3CF*!, T3CFB*
TYPE	T3CPYNF*!
U	T3CF*, T3CF1*!, T3CF2*, T3CFB*!, T3CFCK*, T3CFILE, T3CFSEL*, T3COMP*, T3COP*, T3COPD*, T3CPAK*, T3CPYNF*
>> UP	T3CF*
V	T3CF*, T3CF1!, T3CFB!
X	T3CF*!, T3CF1!, T3CFB!, T3CFSEL, T3COMP*, T3COP*, T3CPAK*, T3CPYNF*
X1	T3CF*!
X1B	T3CF1!, T3CFB!
X1E	T3CF1!, T3CFB!
XB	T3CF1!, T3CF2, T3CFB!, T3CFI!
XE	T3CF1!, T3CF2, T3CFB!, T3CFI!
>> XGLO	T3CFILE*
Y	T3CFB, T3CFSEL, T3COMP, T3COP, T3COPD*!, T3CPYNF
>> YGLO	T3CFILE*

GLOBAL VARIABLES

^DO(T3CF, T3CF2, T3CF1, T3CPAK
^EMPLOY(T3CF, T3CFCK, T3CFILE*
^FM(T3CF*!, T3CF1*!, T3CF2, T3CFB*!
^FORMS(T3CF, T3CF1, T3CF2, T3CFB, T3CFI
^ICC(T3CFCK, T3CFILE*!, T3COMP, T3COP, T3COPD*, T3CPYNF*
^ICCTMP(T3CPAK*!
^INJ(T3COMP*, T3COP*
^ROUTINE(T3CPAK
^UTILITY(T3CPAK*!

ROUTINE

INVOKES:

T3CF	T3CF1, T3CF2, T3CFSEL
T3CF2	%IS, T3CFI
T3CFB	DIC, T3CF2
T3CFCK	hprt`ptrslot2
T3CFI	%IS
T3CFILE	%XY`%RCR, T3COPD
T3CFSEL	DIC, DIE, T2GUSER
T3COMP	%DT
T3COP	%DT
T3CPAK	%XY`%RCR, %RDELETE, R1+3`DIFROM, EMP`T3CPAK
T3CPYNF	%DT

ROUTINE

IS INVOKED BY:

%DT	T3COMP, T3COP, T3CPYNF
%IS	T3CF2, T3CFI
%XY`%RCR	T3CFILE, T3CPAK
%RDELETE	T3CPAK
%DIC	T3CFB, T3CFSEL
%DIE	T3CFSEL
R1+3`DIFROM	T3CPAK
%T2GUSER	T3CFSEL
%T3CF1	T3CF
%T3CF2	T3CF, T3CFB
%T3CFI	T3CF2
%T3CFSEL	T3CF
%T3COPD	T3CFILE
EMP`T3CPAK	T3CPAK
hprt`ptrslot2	T3CFCK

***** END ****

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b RESTRICTIVE MARKINGS NONE	
2a SECURITY CLASSIFICATION AUTHORITY N/A		3 DISTRIBUTION AVAILABILITY OF REPORT Approved for public release; distribution unlimited.	
2b DECLASSIFICATION/DOWNGRADING SCHEDULE N/A		5 MONITORING ORGANIZATION REPORT NUMBER(S)	
4 PERFORMING ORGANIZATION REPORT NUMBER(S) NHRC Report Number. 89-17		7a NAME OF MONITORING ORGANIZATION Commander, Naval Medical Command	
6a NAME OF PERFORMING ORGANIZATION Naval Health Research Center	6b OFFICE SYMBOL (If applicable) Code 20	7b ADDRESS (City, State, and ZIP Code) Department of the Navy Washington, DC 20372	
6c ADDRESS (City, State, and ZIP Code) P.O. Box 85122 San Diego, CA 92138-9174	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER		
8a NAME OF FUNDING/SPONSORING ORGANIZATION Navy Medical Research & Development Command	8b OFFICE SYMBOL (If applicable)	10 SOURCE OF FUNDING NUMBERS	
8c ADDRESS (City, State, and ZIP Code) Navy Medical Command National Capitol Region Bethesda, MD 20814-5044	PROGRAM ELEMENT NO 63706N	PROJECT NO M0095	WORK UNIT ACCESSION NO 005 6050
11 TITLE (Include Security Classification) (U) INJURY AND COMPENSATION CLAIMS MODULE PROGRAMMING MAINTENANCE MANUAL			
12 PERSONAL AUTHOR(S) PEARSALL, D.M.			
13a TYPE OF REPORT FINAL	13b TIME COVERED FROM TO	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
16 SUPPLEMENTARY NOTATION			
17 COSATI CODES		18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP	
		MUMPS, OSHRKS, NOHIMS, OCCUPATIONAL HEALTH, INJURY COMPENSATION AND CLAIMS.	
19 ABSTRACT (Continue on reverse if necessary and identify by block number) This document is to be used by the OSHRKS/NOHIMS site managers in conjunction with other Injury and Compensation Claims Module (ICC) technical InterSystems' M/FORM. A brief description of each package and the routines used to interface the two packages are discussed in this document. Descriptive flow diagrams, data design tables, and programming codes are provided for a better understanding of the ICC Module.			
20 DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> OTC USERS		21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a NAME OF RESPONSIBLE INDIVIDUAL D.M. PEARSALL		22b TELEPHONE (Include Area Code) (619) 553-8471	22c OFFICE SYMBOL Code 20

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

FAB Government Printing Office 1010-007-000